

ABSTRACT

Title of Document: **BAYESIAN INFERENCE WITH
OVERLAPPING DATA: METHODOLOGY
AND APPLICATION TO SYSTEM
RELIABILITY ESTIMATION AND SENSOR
PLACEMENT OPTIMISATION.**

Christopher Stephen Jackson, Doctor of
Philosophy (Ph.D.) in Reliability Engineering
2011

Directed By: **Professor Ali Mosleh, Department of Mechanical
Engineering**

Contemporary complex systems generally have multiple sensors embedded at various levels within their structure. Sensors are data gathering mechanisms that measure a systemic quantity (such as functionality or failure) providing the engineer with a multitude of reliability information. When data sets are drawn simultaneously from multiple sensors in a system, they are said to be overlapping. Current methodologies focus on conducting system reliability analysis of non-overlapping data sets. We introduce a Bayesian methodology that allows analysis of overlapping data sets, exploiting their inherent inter-dependence to yield significant additional information.

Data gathered from a sensor placed at the ‘top’ of the system (i.e. systemic functionality) is contextualised through dependence on data gathered simultaneously

from any sub-system or component. A system that is functional in spite of a non-functional sub-system infers information about the reliability characteristics of the clearly functional system remainder. The same principle extends to any other sensor that has subordinate sensors upon which it is observationally dependent. We apply overlapping Bayesian analysis on several example systems to highlight the information inherent in overlapping data sets and compare these results against previous methodologies that are constrained to non-overlapping data. The differences observed become errors if the incorrect methodology is used.

The overlapping Bayesian methodology we introduce deals with on-demand and continuous life metric systems. The likelihood function for on-demand systems accommodates multiple degraded states and relies on an algorithm we introduce that rapidly generates combinations of disjoint cut-sets that imply the evidence. The likelihood function for continuous life-metric systems (such as those whose failure probability is time based) incrementally examines each sensor data when contextualised through all other data sets. We generalise these likelihood functions for uncertain data, allowing simplification of the likelihood functions through real-life measuring inaccuracies.

Finally, we use the methodologies developed above to assess probable information gain for various sensor placement permutations. We embed this process into a Bayesian experimental design framework that allows sensor placement to be

optimised against information. This can then be fed into any multi-objective optimisation framework, or used in isolation to allow informed sensor placement.

BAYESIAN INFERENCE WITH OVERLAPPING DATA: METHODOLOGY
AND APPLICATION TO SYSTEM RELIABILITY ESTIMATION AND
SENSOR PLACEMENT OPTIMISATION.

By

Christopher Stephen Jackson

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park, in partial fulfilment
of the requirements for the degree of
Doctor of Philosophy
2011

Advisory Committee:
Professor A. Mosleh, Chair
Professor M. Modarres
Professor S. Azarm
Professor A. Christou
Professor G.B. Baecher

© Copyright by
Christopher Stephen Jackson
2011

Preface

This Dissertation has been written in Australian English. The official Australian English Dictionary is the Macquarie Dictionary (www.macquariedictionary.com.au).

Dedication

Developing a dissertation does not come without sacrifices made by both myself and on those around me.

I need to thank Professor Ali Mosleh who went beyond the traditional role accredited to thesis supervisors. Without his help, I would not have been able to complete this document with the challenges associated with the breadth of the Pacific Ocean.

These challenges were mitigated somewhat by the companionship, frivolity and humour of my own two canine intellectuals, Rowdy and Moe, whose unique perspectives (and priorities) on life generally provided somewhat of counterweight against contemporary academia (whether welcome or otherwise).

And finally, my most profound thanks go to my wife Jennifer, who has had to endure much in the development of the work contained herein. Her support has been greatly appreciated, and the fiscal challenges provided by her extreme appetite for shoes has spurred me towards the hopefully higher salaries anticipated on the completion of post-graduate studies.

Acknowledgements

I must acknowledge the Royal Australian Army, which has dominated my professional development and initially propelled me down the path of pursuing post-graduate study at the University of Maryland.

Table of Contents

| | |
|--|-----|
| Preface..... | ii |
| Dedication | iii |
| Acknowledgements..... | iv |
| Table of Contents..... | v |
| List of Tables | x |
| List of Figures | xii |
| Chapter 1: Introduction..... | 14 |
| 1.1. Overlapping and multi-level data in system reliability and risk | 14 |
| 1.2. Research objectives..... | 17 |
| 1.3. Treatment of non-overlapping and <i>overlapping data</i> in Bayes' Theorem . | 17 |
| 1.3.1. Non-overlapping data..... | 18 |
| 1.3.2. <i>Overlapping data</i> | 18 |
| Example 1: Effect of <i>Overlapping data</i> | 19 |
| Example 2: Effect of <i>Overlapping data</i> – Simple time based system | 20 |
| 1.4. <i>Overlapping data</i> analysis of systems | 24 |
| 1.5. Uncertainty in data..... | 26 |
| 1.6. Sensor placement optimisation | 26 |
| Chapter 2: Review of the State of the Art..... | 27 |
| 2.1. Introduction..... | 27 |
| 2.2. Current Methodologies | 28 |
| 2.2.1. Binary-state on-demand systems | 28 |
| Example 3: Non-overlapping data analysis of a binary-state on-demand system | |
| | 30 |
| 2.2.2. Multi-state On-demand Systems..... | 33 |
| Example 4: Development of cut-sets for multi-state systems..... | 36 |
| Example 5: Non-overlapping data analysis of a multi-state on-demand system | 40 |
| 2.3. Aggregation Error | 42 |
| 2.4. Sensor placement optimisation | 43 |
| 2.5. Summary | 44 |
| Chapter 3: Likelihood function of <i>overlapping data</i> of binary-state on-demand systems | 45 |
| 3.1. Introduction..... | 45 |
| 3.2. Likelihood function of binary-state on-demand systems..... | 45 |
| 3.2.1. Step 1: State Vector Analysis | 46 |
| 3.2.2. Step 2: Structure Functions | 46 |
| 3.2.3. Step 3: (Sub-) System Failure Combinations..... | 47 |
| 3.3. Comparison with non-overlapping data methods – Binary-state on-demand systems..... | 50 |
| Example 6: <i>Overlapping data</i> analysis of binary-state on-demand systems | 50 |
| Example 7: Non-overlapping and <i>overlapping data</i> analysis of an on-demand system. | 54 |

| | |
|---|-----|
| Example 8: Non-overlapping and <i>overlapping data</i> analysis of a binary-state on-demand system..... | 57 |
| Example 9: Non-overlapping and <i>overlapping data</i> analysis of a binary-state on-demand system..... | 60 |
| 3.4. Multiple Instances of Identical Components | 62 |
| Example 10: <i>Overlapping data</i> analysis of a binary-state on-demand system with multiple instances of the same component | 63 |
| 3.5. Limitations on previous methodologies..... | 65 |
| Example 11: Limitation of the <i>Overlapping Graves et al. method</i> | 65 |
| 3.6. Summary | 66 |
| Chapter 4: Likelihood function of <i>overlapping data</i> of multi-state on-demand systems | 68 |
| 4.1. Introduction..... | 68 |
| 4.2. Likelihood function of multi-state on-demand systems..... | 68 |
| 4.2.1. Step 1: State Vector Analysis | 68 |
| 4.2.2. Step 2: Structure Functions | 69 |
| 4.2.3. Step 3: (Sub-) System Failure Combinations..... | 70 |
| 4.3. Multiple Instances of Identical Components | 73 |
| 4.4. Comparison with non- <i>overlapping data</i> methods – Multi-state on-demand systems..... | 74 |
| Example 12: <i>Overlapping data</i> analysis of a multi-state on-demand system..... | 74 |
| 4.5. Evaluation | 80 |
| 4.6. Summary | 80 |
| Chapter 5: Downwards Inference Algorithm: Evaluation of <i>overlapping data</i> likelihood function of on-demand systems | 82 |
| 5.1. Introduction..... | 82 |
| 5.2. Multi-state on-demand complex systems: Likelihood Function..... | 82 |
| 5.3. Mathematical Representation: Evidence as a function of <i>sensor information vectors</i> | 86 |
| Example 13: Matrix equation of <i>sensor information vectors</i> | 88 |
| 5.4. Downwards Inference – Combination Generation Algorithm..... | 90 |
| 5.4.1. Step 1: Generation of Constraints..... | 91 |
| 5.4.2. Step 2: Identification of common v_i^S values (optional)..... | 91 |
| 5.4.3. Step 3: Setting minimum and maximum v_i^S values..... | 94 |
| 5.4.4. Step 4: Generation of combinations - ‘Sideways consideration / Upwards generation’ | 96 |
| Example 14: Solving matrix equation of <i>sensor information vectors</i> | 99 |
| 5.5. Downwards Inference - Cut Set Generation Algorithm | 103 |
| 5.5.1. Step 1: Generation of Constraints..... | 104 |
| 5.5.2. Step 2: Setting minimum and maximum component state variables, x_j | 105 |
| 5.5.3. Step 3: Generation of cut sets - ‘Sideways consideration / Upwards generation’ | 107 |
| Example 15: Cut-set (state vector) generation for <i>sensor information vectors</i> | 108 |
| 5.6. Evaluating Downwards Inference Likelihood function..... | 113 |
| Example 16: Multi-state on-demand system likelihood function evaluation ... | 113 |

| | |
|---|-----|
| 5.7. Summary | 115 |
| Chapter 6: <i>Overlapping data</i> likelihood function of systems with continuous life metrics | 117 |
| 6.1. Introduction..... | 117 |
| 6.2. Time based failure probability | 117 |
| Example 17: Inference Diagrams..... | 120 |
| 6.3. Hierarchy of sensors and components | 125 |
| 6.4. Probability density functions of times that sensors detect failure..... | 129 |
| 6.5. Censored data..... | 136 |
| 6.6. Continuous life metric system - algorithm..... | 138 |
| 6.6.1. Step 1: Determine the set of all inferentially subordinate components and sensors for each sensor..... | 139 |
| 6.6.2. Step 2: Model system failure detection probabilities on inferentially subordinate components and sensors..... | 139 |
| 6.6.3. Step 3: Compile evidence. | 140 |
| 6.6.4. Step 4: Compile evidential probability of failure detection p_i^S for all inferentially subordinate sensors..... | 140 |
| 6.6.5. Step 5: Differentiate all sensor failure detection probabilities, p_i^S , with respect to <i>inferentially subordinate</i> sensor failure detection probabilities, p_i^S | 140 |
| 6.6.6. Step 6: Differentiate all sensor failure detection probabilities, p_i^S , with respect to <i>inferentially subordinate</i> component failure probabilities, p_j | 141 |
| 6.6.7. Step 7: For all sensors, compile set of all <i>inferentially subordinate</i> sensors that have identical failure detection times..... | 141 |
| 6.6.8. Step 8: Substitute all elements into the likelihood function. | 141 |
| Example 18: Basic two component continuous life metric system <i>overlapping data</i> analysis..... | 141 |
| Example 19: Basic two component continuous life metric system <i>overlapping data</i> analysis..... | 145 |
| Example 20: Basic two component continuous life metric system <i>overlapping data</i> analysis..... | 148 |
| 6.7. Summary | 154 |
| Chapter 7: Uncertain Evidence | 156 |
| 7.1. Introduction..... | 156 |
| 7.2. General framework for uncertain data | 156 |
| 7.3. Evidence uncertainty for on-demand systems. | 158 |
| 7.4. Continuous Time Based Systems..... | 159 |
| 7.4.1. Step 1: Determine the set of all inferentially subordinate components and sensors for each sensor..... | 161 |
| 7.4.2. Step 2: Model system failure detection probabilities on inferentially subordinate components and sensors..... | 161 |
| 7.4.3. Step 3: Compile uncertain evidence. | 162 |
| 7.4.4. Step 4: Compile evidential probability of failure detection p_i^S for all inferentially subordinate sensors..... | 162 |

| | | |
|------------|--|-----|
| 7.4.5. | Step 5: Substitute all elements into the likelihood function. | 162 |
| | Example 21: <i>Overlapping data</i> analysis of a continuous time-based system with inherent timing inaccuracies. | 163 |
| 7.5. | Summary | 167 |
| Chapter 8: | Sensor Placement: Maximising information from Bayesian analysis of complex systems | 168 |
| 8.1. | Introduction..... | 168 |
| 8.2. | Bayesian Experimental Design | 169 |
| 8.3. | Utility Function | 172 |
| 8.4. | Information Optimization through information utility functions..... | 174 |
| 8.4.1. | Fisher Information. | 175 |
| 8.4.2. | Shannon Information. | 175 |
| 8.5. | Information Utility Function..... | 177 |
| 8.5.1. | Information of the Posterior Distribution. | 178 |
| 8.5.2. | Information difference of Posterior/Prior distributions. | 178 |
| 8.5.3. | Information of specific parameters. | 178 |
| 8.5.4. | Information of variables that are functions of parameters. | 179 |
| 8.6. | Sensor Placement | 180 |
| 8.6.1. | Prior Information. | 180 |
| 8.6.2. | Available Sensor Locations. | 181 |
| 8.6.3. | Information Utility. | 181 |
| 8.6.4. | System Logic (Bayesian Analysis). | 181 |
| 8.6.5. | Nature of the evidence. | 182 |
| 8.6.6. | Deriving structure functions. | 182 |
| 8.6.7. | Simulation of evidence for on-demand systems. | 183 |
| 8.6.8. | Simulation of evidence for continuous time based systems. | 186 |
| 8.6.9. | Simulation of posterior distributions. | 188 |
| 8.6.10. | Expected Information Utility. | 188 |
| | Example 22. Expected information utility for various sensor placement arrangements for an on-demand system. | 189 |
| | Example 23. Expected information utility for various sensor placement arrangements for a continuous time-based system. | 194 |
| 8.7. | Summary | 199 |
| Chapter 9: | Conclusion | 201 |
| 9.1. | Overlapping-Data..... | 201 |
| 9.2. | Overlapping data Analysis | 202 |
| 9.2.1. | Binary-state on-demand systems | 202 |
| 9.2.2. | Multi-state on-demand systems | 203 |
| 9.2.3. | Overlapping data analysis of on-demand systems: Algorithm | 203 |
| 9.2.4. | Continuous Life Metric Systems | 204 |
| 9.3. | Bayesian analysis of uncertain data | 204 |
| 9.3.1. | On-demand systems | 205 |
| 9.3.2. | Continuous time-based systems..... | 205 |
| 9.4. | Sensor Placement: Maximising information from Bayesian analysis of complex systems | 205 |
| 9.4.1. | Bayesian Experimental Design | 206 |

| | |
|------------------------|-----|
| 9.5. Further work..... | 206 |
| Appendix A..... | 208 |
| Bibliography | 211 |

List of Tables

| | |
|--|-----|
| Table 1: Scope of inference on system (component 2)..... | 20 |
| Table 2: Expression of system level evidence for the Hamada et al method | 29 |
| Table 3: An example of possible states of components 1 to 5 of a given multi-state system | 34 |
| Table 4: Expression of system level evidence for the Hamada et al method | 40 |
| Table 5: State Vectors of system in Figure 1..... | 51 |
| Table 6: Possible state vector combinations of system in Figure 1. | 52 |
| Table 7: Truncated list of <i>state vectors</i> of system in Figure 1..... | 75 |
| Table 8: Possible state vector combinations of system in Figure 1. | 76 |
| Table 9: Comparison of data inference when data is constrained to be non-overlapping (Graves et al. method) versus overlapping (downwards inference) for analysis of system examined in Example 12. | 79 |
| Table 10: List of relevant <i>sensor information vectors</i> for the system in Figure 30 with evidence in equations (95), (96) and (97). | 109 |
| Table 11: List of relevant <i>sensor information vectors</i> for the system in Figure 30 with evidence in equations (95), (96) and (97) and mutually exclusive cut-sets..... | 114 |
| Table 12: <i>Component numbers</i> and <i>component type numbers</i> for components in the system illustrated in figure 16..... | 120 |
| Table 13: Expression of multi-level evidence for binary-state on-demand systems | 127 |
| Table 14: Component reliability characteristics for system illustrated in Figure 44. | 142 |
| Table 15: Sets of <i>inferentially subordinate</i> sensors and components for the system in Figure 44. | 142 |
| Table 16: Partial derivatives of higher level failure probabilities for the missile guidance system..... | 143 |
| Table 17: Missile Guidance System component reliability characteristics for time expressed as minutes (* - Jeffreys' non-informative prior distributions for relevant parameters [19]). | 149 |
| Table 18: Sets of <i>inferentially subordinate</i> sensors and components for the system in Figure 47 | 150 |
| Table 19: Partial derivatives of higher level failure probabilities for the missile guidance system..... | 151 |
| Table 20: Missile Guidance System reliability parameter statistics (derived from Markov chain Monte Carlo simulation with 100 000 draws). | 154 |
| Table 21: Examples of information based <i>Utility Functions</i> of experimental frameworks | 179 |
| Table 22: All possible <i>component state vectors</i> with probabilities of occurrence ... | 191 |
| Table 23: All possible <i>sensor information vectors</i> with probabilities of occurrence | 192 |
| Table 24: Possible evidence sets with probabilities of occurrence (10 most probable of 90 evidence sets simulated) | 193 |
| Table 25: Expected information utility for each sensor arrangement..... | 194 |
| Table 26: Simulated failure detection times based on current state of knowledge (five most probable evidence sets displayed ... based on 100 000 simulations) | 198 |

| | |
|---|-----|
| Table 27: Expected information utility for each sensor arrangement..... | 198 |
|---|-----|

List of Figures

| | |
|--|----|
| Figure 1: Basic two component series system | 19 |
| Figure 2: Reliability block diagram of power module..... | 21 |
| Figure 3: Posterior distribution of sub-system failure rates of parallel system treating data as overlapping and non-overlapping. | 24 |
| Figure 4: Joint posterior distribution of p_1 and p_2 - <i>Test 2</i> | 32 |
| Figure 5: Joint posterior distribution of p_1 and p_2 - <i>Test 1</i> | 33 |
| Figure 6: Multi-state gate state variable relationships | 35 |
| Figure 7: Basic three component system with a ‘2 out of 3’ gate defining the top event | 36 |
| Figure 8: Multi-state two component series system | 40 |
| Figure 9: Marginal posterior distributions of each component state probability of system components in Figure 8 (using the Graves et al <i>non-overlapping</i> method).... | 42 |
| Figure 10: Joint posterior distributions of p_1 and p_2 from <i>Test 1</i> | 52 |
| Figure 11: Percentage error in <i>non-overlapping data</i> likelihood function (maximum normalised error = 3.55 %) | 53 |
| Figure 12: <i>Overlapping data</i> Evidence..... | 54 |
| Figure 13: <i>Non-overlapping data</i> Evidence..... | 55 |
| Figure 14: Joint posterior distributions of p_1 and p_2 from test 3 evidence..... | 56 |
| Figure 15: Percentage error in <i>non-overlapping data</i> likelihood function (maximum error = 59.97 %)..... | 56 |
| Figure 16: <i>Overlapping data</i> Evidence..... | 58 |
| Figure 17: <i>Non-overlapping data</i> Evidence | 58 |
| Figure 18: Joint posterior distributions of p_1 and p_2 from test 4 evidence..... | 59 |
| Figure 19: Percentage error in <i>non-overlapping data</i> likelihood function (maximum error = 24.99 %)..... | 59 |
| Figure 20: Data Inference | 61 |
| Figure 21: Normalized likelihood functions of p_1 and p_2 from test 1 evidence (identical) | 61 |
| Figure 22: Basic two identical component series system | 63 |
| Figure 23: <i>Overlapping data</i> Evidence..... | 64 |
| Figure 24: Identical Normalized likelihood functions of p_1 from test 5 evidence | 65 |
| Figure 25: Three sensor on-demand system with <i>paths of apparent influence</i> indicated | 66 |
| Figure 26: Marginal posterior distributions of each component state probability of the system in Figure 1 (the distributions coloured grey are those derived by the Graves et al. method illustrated in Example 5). | 77 |
| Figure 27: Evidence inferred information when considered as non-overlapping..... | 77 |
| Figure 28: Evidence inferred information when considered as overlapping | 78 |
| Figure 29: Downwards Inference – Likelihood Function Evaluation Process | 85 |
| Figure 30: Multi-state two component series system | 88 |
| Figure 31: Representation of ‘sideways consideration / upwards generation’ combination algorithm | 97 |

| | |
|--|-----|
| Figure 32: Downwards Inference Combination Generation Algorithm Flow-Chart (‘sideways consideration / upwards generation’) | 98 |
| Figure 33: Generation of <i>sensor information vector combinations</i> | 103 |
| Figure 34: Overlapping Cut Set Generation Algorithm (‘sideways consideration / upwards generation’) | 108 |
| Figure 35: Generation of <i>component cut-sets</i> | 112 |
| Figure 36: Three sensor on-demand system with <i>paths of apparent influence</i> indicated | 121 |
| Figure 37: Three <i>apparent sub-systems</i> based on inference relationships | 122 |
| Figure 38: Three <i>apparent sub-systems</i> based on inference relationships | 124 |
| Figure 39: Hierarchy of a 5 component, 5 sensor system <i>with respect to sensor #2</i> | 126 |
| Figure 40: Representation of the probabilistic relationship between component failure times and times to detection of failure by sensors | 130 |
| Figure 41: Representation of the probabilistic relationship between component failure times and sensor failure detection times – sensor inference sub-systems | 130 |
| Figure 42: Representation of the probabilistic relationship between component failure times and times at which sensors detect failure – sensor inference sub-systems (where the times to failure detection of sensors #2 and #3 are known) | 132 |
| Figure 43: Conditional CDFs of time to failure detection of sensors #2 and #1 respectively, given the failure detection time of subordinate sensors | 133 |
| Figure 44: Basic two component parallel system | 142 |
| Figure 45: Likelihood function for system illustrated in Figure 44 with evidence set (151) | 145 |
| Figure 46: Likelihood function for system illustrated in Figure 44 with evidence set (155) | 148 |
| Figure 47: Missile Guidance System | 149 |
| Figure 48: Likelihood functions of system illustrated in Figure 44 with evidence sets (151) and (155), analysed with an uncertainty in measurement of 0,001 | 165 |
| Figure 49: 4 component (3 component type) binary-state on-demand system | 189 |
| Figure 50: Marginal distributions of the unknowns of interest generated by our current state of knowledge (prior distributions from equations (226), (227), (228) and evidence set (229)) | 195 |
| Figure 51: Time to failure distributions for components 1, 2, 3 and 4 based on the state of knowledge represented in Figure 50 | 196 |

Chapter 1: Introduction

1.1. OVERLAPPING AND MULTI-LEVEL DATA IN SYSTEM RELIABILITY AND RISK

It is difficult to imagine contemporary complex systems (ranging from personal motor vehicles to nuclear power plants) that do not have multiple sensors embedded at various levels within their structure. Sensors are data gathering mechanisms that measure a systemic quantity, such as functionality or failure in the context of system reliability analysis. Technology has advanced to a stage where these sensors can be implanted relatively cheaply and effectively, thus providing the engineer with a multitude of functionality and reliability information. However, data sets gathered simultaneously from multiple sensors within the same system are unique: *they are overlapping*.

Consider a system where a particular sensor has detected that a sub-system has failed. Without any further information (and from a diagnostics perspective only), inference can only be made about the ‘unreliability’ of that sub-system. If another sensor simultaneously detects that the entire system is functional, additional information is at hand: it is suggested that the remainder of the system is still functioning to an extent that mitigates the sub-system failure. From a prognostics perspective, system failure can now be more readily predicted as it is now completely dependent on the functionality of the remainder of the system only. But there is now information about the ‘reliability’ of the remainder of the system: information that when correctly inferred improves our understanding of systemic reliability.

Correctly analysing these data sets, with their inherent dependencies, yields significantly more information. The inference made from data drawn from the system level sensor is dependent on data drawn from the sub-system level when drawn simultaneously. System functionality is conditional on the functionality of subordinate sub-systems and components. In this way, data sets drawn simultaneously from multiple sensors from the same system are *overlapping* and contain information through their inherent inter-dependencies and require specific analysis techniques.

To instigate formal definition, sets of *overlapping data* are those that meet the following criteria: simultaneity (the sets are drawn from observations or demands that occur at the same time); and correspondence (the sets are dependent on the same system or process). A common example of *overlapping data* is that already introduced where reliability data is drawn from a particular system through multiple sensors simultaneously. At many points throughout this dissertation, comparisons are made between analysis of data sets as though they were either *overlapping* or *non-overlapping* to expose extremely significant differences. It is therefore crucial for the reliability engineer to understand what *overlapping data* is and how to analyse it.

To date, the majority of system reliability data analysis methodologies can only analyse *non-overlapping data*. A further observation of these techniques is that since system reliability is a function of component reliabilities, system reliability analysis has been focussed on reliability testing at the component level. These approaches automatically preclude useful system and sub-

system data (which is referred to as higher level data as it appears ‘higher’ in many visualization methodologies).

Reliability analysis involves the ‘downwards’ propagation of information: test data can infer information ‘down’ into underlying reliability parameters (the term ‘down’ is used as it is common for component, sub-system and then system levels to be graphically represented hierarchically above reliability parameters). Reliability prediction involves ‘upwards’ propagation of information since reliability parameters (through reliability models and system logic) define higher-level performance. Component level testing is straightforward in that information can be propagated directly downwards into component reliability parameters. System and sub-system level data relies on system logic when information is propagated downwards, making analysis more complicated, especially when data is *overlapping*. This complexity presents not only challenges but opportunities: the fact that sensors embedded in systems infer information about many components can be judiciously exploited by the test engineer to maximise information gain.

1.2. RESEARCH OBJECTIVES

The fundamental research objectives of this dissertation are as follows:

- a. To develop Bayesian frameworks and methodologies to allow analysis of *overlapping data* for (both binary and multi-state) on-demand systems and continuous life metric systems;
- b. To develop mathematical techniques and algorithms for rapid evaluation of all steps in the above methodologies;
- c. To generalise the above methodologies to incorporate uncertain data; and
- d. To combine the above methodologies with optimisation techniques to refine sensor placement within complex systems.

1.3. TREATMENT OF NON-OVERLAPPING AND *OVERLAPPING DATA* IN BAYES' THEOREM

Bayes' theorem is written formally as:

$$\pi_1(\boldsymbol{\theta} | E) = \frac{L(E | \boldsymbol{\theta})\pi_0(\boldsymbol{\theta})}{\int_{\forall \boldsymbol{\theta}'} L(E | \boldsymbol{\theta}')\pi_0(\boldsymbol{\theta}')d\boldsymbol{\theta}'} \dots^1 \quad \text{---(1)}$$

where $\boldsymbol{\theta}$ is the set of *unknowns of interest* or parameters, $\pi_0(\boldsymbol{\theta})$ is the prior distribution of $\boldsymbol{\theta}$ (representing the initial state of knowledge), $L(E | \boldsymbol{\theta})$ is the likelihood of observing a set of evidence, E , for a given $\boldsymbol{\theta}$, and $\pi_1(\boldsymbol{\theta} | E)$ is the updated posterior distribution of the set of *unknowns of interest* or parameters representing the updated state of knowledge.

¹ Throughout this dissertation, any generic evidence set is denoted E .

1.3.1. Non-overlapping data.

Non-overlapping data sets are ‘observationally’ independent and therefore generate independent likelihood functions. In the context of a complex system of n components where the vector, $\tilde{\mathbf{p}}_j$, that contains the probabilities of the j^{th} component being in a particular functional state, and the set or matrix of n state probability sets for the system is $\mathbf{p} = \{\tilde{\mathbf{p}}_1, \tilde{\mathbf{p}}_2, \tilde{\mathbf{p}}_3, \dots, \tilde{\mathbf{p}}_j, \dots, \tilde{\mathbf{p}}_n\}$, Bayes’ theorem becomes:

$$\pi_1(\mathbf{p} | E = \{E_1, E_2, E_3, \dots\}) = \frac{L(E_1 | \mathbf{p}) L(E_2 | \mathbf{p}) \times \dots \times \pi_0(\mathbf{p})}{\int_{\forall \mathbf{p}} L(E_1 | \mathbf{p}') L(E_2 | \mathbf{p}') \times \dots \times \pi_0(\mathbf{p}') d\mathbf{p}'} \dots^2 \quad \text{---(2)}$$

where $\{E_1, E_2, E_3, \dots\}$ is a number of *non-overlapping data*/evidence sets.

1.3.2. Overlapping data

The likelihood functions for *overlapping data* are interdependent, and cannot be substituted into equation (2). They generate one encompassing likelihood function. *Overlapping data* is represented formally in Bayes’ theorem as:

$$\pi_1(\mathbf{p} | E = \{E_1^\bullet, E_2^\bullet, E_3^\bullet, \dots\}) = \frac{L(\{E_1^\bullet, E_2^\bullet, E_3^\bullet, \dots\} | \mathbf{p}) \pi_0(\mathbf{p})}{\int_{\forall \mathbf{p}} L(\{E_1^\bullet, E_2^\bullet, E_3^\bullet, \dots\} | \mathbf{p}') \pi_0(\mathbf{p}') d\mathbf{p}'} \quad \text{---(3)}$$

where $\{E_1^\bullet, E_2^\bullet, E_3^\bullet, \dots\}$ is a number of *overlapping data*/evidence sets.

² Note that throughout this dissertation, a convention where a vectors are represented as $\tilde{\mathbf{x}}$, and matrices as \mathbf{x} will be observed. In this instance, \mathbf{p} is a vector but will be annotated as a matrix to align with multi-state system nomenclature that will become apparent further on.

As *overlapping data* sets are dependent, the overall likelihood function is not a product of individual data set likelihood functions.

$$L\left(E = \{E_1^\bullet, E_2^\bullet, E_3^\bullet, \dots\} | \mathbf{p}\right) \neq L\left(E_1^\bullet | \mathbf{p}\right) \times L\left(E_2^\bullet | \mathbf{p}\right) \times L\left(E_3^\bullet | \mathbf{p}\right) \times \dots \quad \text{---(4)}$$

Effects of inter-dependent likelihood functions are examined in Example 1 and Example 2.

Example 1: Effect of *Overlapping data*

Consider the basic two component series system examined in Figure 1 that is to a test regime 1:

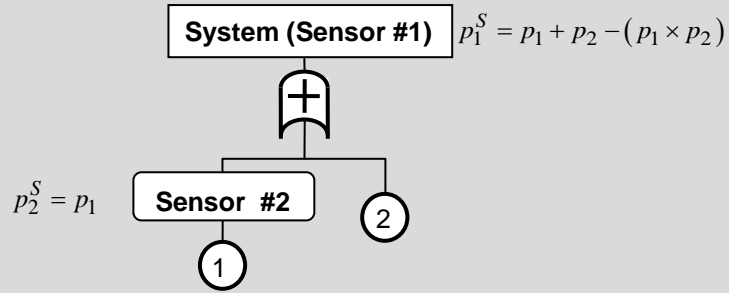


Figure 1: Basic two component series system

Test 1. A series of 10 demands where 10 failures were detected at the system level and 1 failure was detected by sensor #2.

The effect of observing different numbers of failure at sensor #2 (for 10 systemic demands) are explored in Table 1 within the context of systemic failure on every demand. It can be seen that the number of failures detected by sensor #2 affects the level of inference that the number of systemic failures has on the failure characteristics of component 2.


| No of failures detected by sensor #1 k_1^S | No of failures detected by sensor #2 k_2^S | No of possible instances of component 2 failing | Uncertainty in the behaviour of component 2 |
|---|---|---|---|
| 10 | 0 | 10 | Specific |
| | 1 | 9,10 | (no uncertainty) |
| | 2 | 8, 9,10 |  |
| | 3 | 7, 8, 9,10 | |
| | 4 | 6, 7, 8, 9,10 | |
| | 5 | 5, 6, 7, 8, 9,10 | |
| | 6 | 4, 5, 6, 7, 8, 9,10 | |
| | 7 | 3, 4, 5, 6, 7, 8, 9,10 | |
| | 8 | 2, 3, 4, 5, 6, 7, 8, 9,10 | |
| | 9 | 1, 2, 3, 4, 5, 6, 7, 8, 9,10 | |
| | 10 | 0, 1, 2, 3, 4, 5, 6, 7, 8, 9,10 | Ambiguous (total uncertainty) |

Table 1: Scope of inference on system (component 2)

Example 2: Effect of *Overlapping data* – Simple time based system

Consider a simple power module that involves two different and parallel transformer/filter sub-systems. The system is represented in the reliability block diagram in Figure 2. There are two sensors that detect time to failure: sensor #1 for the entire system, and sensor #2 for sub-system A. It is assumed that the time to failure for each sub-system is exponentially distributed, with the prior distributions of μ_A and μ_B be (the means of each sub-systems' time to failure) are uniformly distributed across [0,20] months.

i.e. $\theta = \{\mu_A, \mu_B\}$ where θ is the set of all system reliability parameters. ---(5)

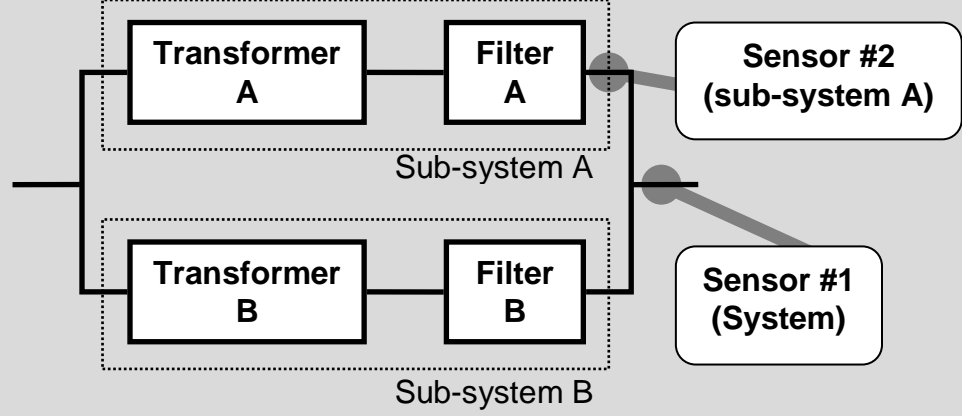


Figure 2: Reliability block diagram of power module

Similarly, the respective probability density functions (*PDFs*) and cumulative distribution functions (*CDFs*) for the system and sub-systems are $f(t \mid \theta)$, $f_A(t \mid \mu_A)$, $f_B(t \mid \mu_B)$ and $F(t \mid \theta)$, $F_A(t \mid \mu_A)$, $F_B(t \mid \mu_B)$ where t is time. The system *CDF* and *PDF* are functions of the sub-systems' *CDFs* and *PDFs*.

$$F = F_A \bullet F_B \quad \text{---(6)}$$

as the system is a parallel configuration of sub-systems A and B , making systemic failure probability a product of sub-system failure probabilities

$$f = f_A F_B + f_B F_A \dots \text{derivative of equation (6) with respect to } t \quad \text{---(7)}$$

Two evidence/data sets, in (8) and (9), are gathered from each sensor from three tests.

$$E_1 = \tilde{t}_1^S = \{11.12, 6.99, 2.25\} \dots \text{observed failure detection times of sensor \#1} \quad \text{---(8)}$$

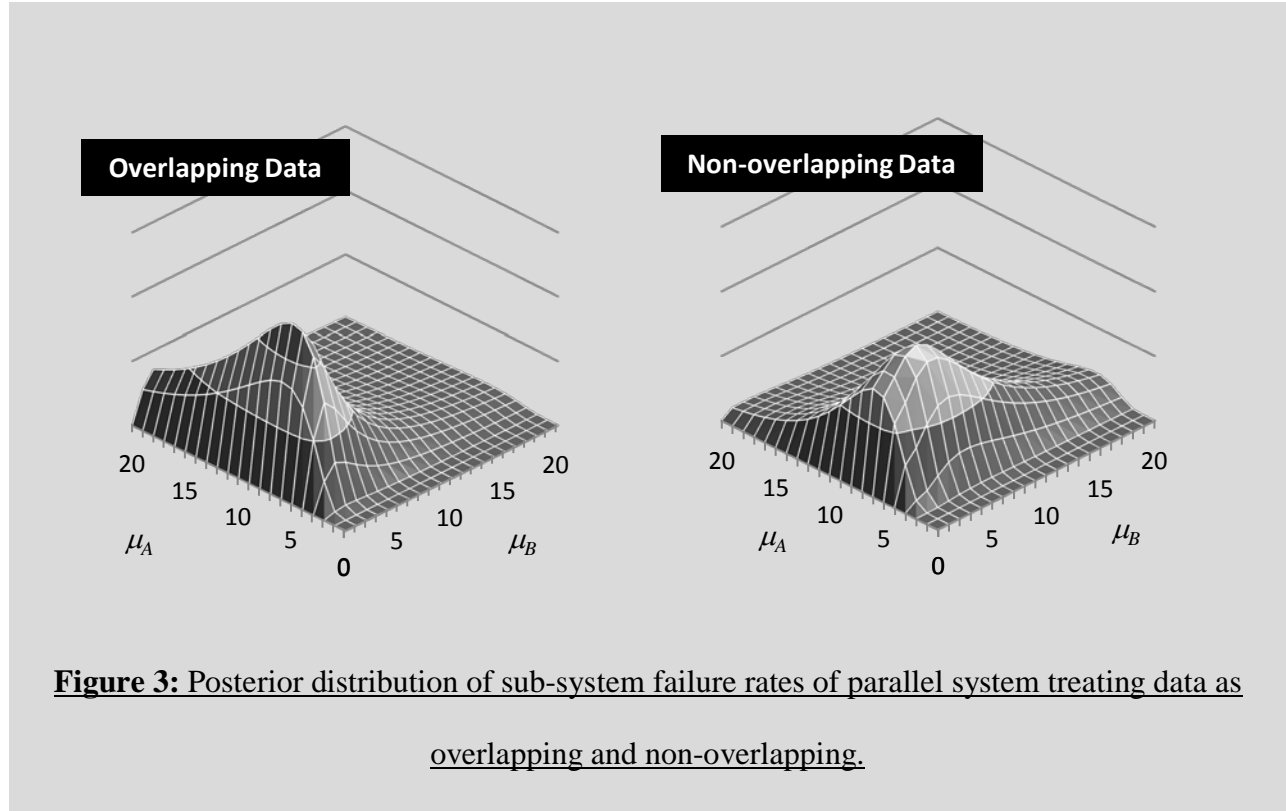
$$E_2 = \tilde{t}_2^S = \{11.12, 6.99, 2.23\} \dots \text{observed failure detection times of sensor \#2} \quad \text{---(9)}$$

Scenario 1: Treating the data sets in (8) and (9) as non-overlapping. This scenario is equivalent to testing the system in isolation three times and observing evidence set (8), and subsequently testing sub-system A in isolation three times and observing evidence set (9). Consequently, the analysis is unsure at what time sub-system A failed when system level evidence is gathered. The likelihood function of observing the evidence sets is based on equations (6) and (7):

$$\begin{aligned} L(E|\theta) &= L(\tilde{t}_1^S, \tilde{t}_2^S | F_A, F_B) = \left[\prod_{\forall t_1^S} \left(\frac{t_1^S}{\mu_A} \cdot \frac{t_1^S}{\mu_B} \right) \right] \cdot \left[\prod_{\forall t_2^S} \left(\frac{t_2^S}{\mu_A} \right) \right] \\ &= \left[\prod_{\forall t_1^S} \frac{e^{-\frac{t_1^S}{\mu_A}}}{\mu_A} \left(1 - e^{-\frac{t_1^S}{\mu_B}} \right) + \frac{e^{-\frac{t_1^S}{\mu_B}}}{\mu_B} \left(1 - e^{-\frac{t_1^S}{\mu_A}} \right) \right] \cdot \left[\prod_{\forall t_2^S} \frac{e^{-\frac{t_2^S}{\mu_A}}}{\mu_A} \right] \\ &= \frac{e^{-\frac{1}{\mu_A} \sum_{\forall t_2^S} t_2^S}}{(\mu_A)^3} \prod_{\forall t_1^S} \frac{e^{-\frac{t_1^S}{\mu_A}}}{\mu_A} + \frac{e^{-\frac{t_1^S}{\mu_B}}}{\mu_B} - \left(\frac{1}{\mu_A} + \frac{1}{\mu_B} \right) e^{-\left(\frac{1}{\mu_A} + \frac{1}{\mu_B} \right) t_1^S} \\ &= \frac{e^{-\frac{1}{\mu_A} \left(\frac{11.12}{\mu_A} + \frac{6.99}{\mu_A} + \frac{2.23}{\mu_A} \right)}}{(\mu_A)^4 \mu_B} \cdot \left(\begin{aligned} &\left(\mu_B e^{-\frac{11.12}{\mu_A}} + \mu_A e^{-\frac{11.12}{\mu_B}} - (\mu_B + \mu_A) e^{-\frac{11.12}{\mu_A} + \frac{1}{\mu_B}} \right) \\ &\cdot \left(\mu_B e^{-\frac{6.99}{\mu_A}} + \mu_A e^{-\frac{6.99}{\mu_B}} - (\mu_B + \mu_A) e^{-\frac{6.99}{\mu_A} + \frac{1}{\mu_B}} \right) \\ &\cdot \left(\mu_B e^{-\frac{2.25}{\mu_A}} + \mu_A e^{-\frac{2.25}{\mu_B}} - (\mu_B + \mu_A) e^{-\frac{2.25}{\mu_A} + \frac{1}{\mu_B}} \right) \end{aligned} \right) \quad \text{---(10)} \end{aligned}$$

Scenario 2: Treating the data sets in (8) and (9) as overlapping. If the data sets (8) and (9) were treated as *overlapping*, it would be equivalent to testing the system three times only and simultaneously observing the evidence sets. Considering the first test where both sensors detected failure at the same time (11.12 months), it can be concluded that sub-system *B* failed sometime before 11.12 months, with sub-system *A* subsequently failing at 11.12 months causing the entire system to fail (noting that it is a parallel system). The same principle is applied for the test where failure was detected by both sensors at 6.99 months in the second test. The third test involved sensor #2 detecting failure at 2.23 months, slightly before sensor #1 detects failure (at 2.25 months). It can be concluded that in this test, sub-system *A* failed at 2.23 months, while sub-system *B* failed at 2.25 months causing the entire system to fail. This illustrates the nature of the information inherent in *overlapping data sets*.

The posterior distributions generated by Bayesian analysis treating the data sets as *overlapping* and *non-overlapping* are illustrated in Figure 3. The techniques required for analysing the data set as if it were *overlapping* are developed later in this dissertation. It can be seen that the primary difference between each posterior distribution is the information pertaining to sub-system *B*: the maximum likelihood estimate (MLE) of μ_B when treating the data as *non-overlapping* (3.98) is approximately double the MLE of μ_B when treating the data as *overlapping* (1.99). Conversely, the MLE of μ_A increases slightly (from 5.99 to 6.78). It is clear that *overlapping data* does not infer the same information when it is analysed as *non-overlapping data*. The difference in information can sometimes be significant, and hence needs to be treated accordingly.



1.4. OVERLAPPING DATA ANALYSIS OF SYSTEMS

There are various systems that can be studied by the reliability engineer. On-demand systems are those that are subject to discrete demands or trials and will respond by operating (or existing) within certain discrete states. Binary-state on-demand systems have only two such states; failure and success (or functionality). By definition, on-demand systems are made up of on-demand components that also operate in the same two discrete states (noting that the maximum number of states each component can exist in is the number of system level states). The state that an on-demand system or component exists in is a discrete random variable. The probability of the state of an on-demand system is a function of the component state probabilities, dictated by the system structure or logic. A methodology for the analysis of *overlapping data sets* for binary-state on-demand systems is developed in Chapter 3.

Multi-state systems are those where components can exist in states ranging incrementally in severity from ‘functional’ to ‘failed’. There will be one or more states of degraded functionality that a component can exist in. A binary-state system is the simplest form of a multi-state system. The method developed in Chapter 3 for the analysis of *overlapping data sets* generated by binary-state on-demand systems is generalized in Chapter 4 to accommodate multi-state systems.

Both of the above methodologies require the generation of sets of combinations of state vectors that imply the evidence. The trivial approach to developing these sets (where each set is considered sequentially) is extremely computationally intensive, with huge numbers of possible combinations requiring examination. An algorithm is developed in Chapter 5 that rapidly compiles these sets, supporting the methodologies developed in Chapters 3 and 4.

Systems and components with continuous life metrics (such as time) contrast from their on-demand counterparts in that the probability that they exist in a particular state (or transition from one state to another) is a function of said life metric. The point at which a component or system transitions from one state to another (measured in terms of the life metric) is a continuous random variable. There are numerous systems that are based on other continuous variables beyond time (such as distance and flow), and the methodology developed in Chapter 6 to analyse *overlapping data sets* (even though it is exclusively are based on time) is equally transferrable to such variables.

1.5. UNCERTAINTY IN DATA

A constant reality of data gathering is uncertainty. Chapter 7 deals with uncertainty in *overlapping data sets*, and examines how the methodologies developed in Chapters 3 to 6 can be modified to accommodate measurement error. A useful development for the case of systems with continuous life metrics is that the likelihood function required to deal with *overlapping data sets* is simplified significantly, and is a reality experienced by every system in this context.

1.6. SENSOR PLACEMENT OPTIMISATION

System sensor placement can come at significant resource costs, and possible sensor locations can be limited by the operating environment. The placement of sensors based on information optimality is necessarily complicated by many factors. By considering the issue of sensor placement as a holistic multi-objective optimization problem, sensor placement can be formally addressed.

Bayesian analysis of *overlapping data* drawn from multi-sensor systems enables the concept of experimental design to be used to optimise sensor placement. The ability to correctly incorporate information inherent in *overlapping data sets* allows precise sensor placements to be analysed in terms of expected information gain. Chapter 8 deals with sensor placement optimisation in a Bayesian experimental design framework, with each permutation of possible sensor locations becoming a separate and distinct experiment. This allows informed decisions to be made on what can ultimately be costly decisions to install sensors at various places within a system or process.

Chapter 2: Review of the State of the Art

2.1. INTRODUCTION

Current system reliability analysis methodologies focus on using system failure logic (such as that represented in reliability block diagrams or fault-trees) to express system failure probability based on the failure probabilities of subordinate components. For example, the Nuclear Regulatory Commission (NRC) Probabilistic Risk Assessment Guide breaks this process into defining component failure probabilities, determining system minimal cut-set probabilities and using them to quantify system reliability [1]. This procedure is replicated in many other relevant textbooks and handbooks.

Higher level data is generally gathered by sensors placed throughout the system structure (a sensor can be a dedicated device or a person: the driver of a car will be instantly aware of the time at which it ceases to function). Whilst a sensor typically always exists at the system level (as in system failure can be immediately observed or detected), it may be desirable to place sensors elsewhere in the system to gather more diagnostic information. Each sensor does not form part of the system function, but can detect whether the system is functional at that point. Sensors are most applicable for sub-systems with their own functionality with failure becoming easily detected. This chapter is a review of current methodologies that analyse data drawn from multi-sensor systems.

2.2. CURRENT METHODOLOGIES

2.2.1. Binary-state on-demand systems

The *state variable* for the j^{th} component of a binary-state on-demand system, x_j , is formally defined in equation (11).

$$x_j = \begin{cases} 0 & \text{if the } j^{\text{th}} \text{ component is functional} \\ 1 & \text{if the } j^{\text{th}} \text{ component has failed} \end{cases} \quad \text{---(11)}$$

An approximate Bayesian method was considered through reliability block diagram methodology that combined *non-overlapping* system and component level data by Mastran and Singpurwalla [2]. A top-down approach combined system level data with component level data and prior distributions to update component life characteristics, which in turn provided a posterior distribution of system reliability. An alternate approximate Bayesian procedure based on a bottom-up approach with prior distribution parameters which were then combined with data was developed by Martz et al [3] and Martz and Waller [4].

Fully Bayesian techniques were developed by Johnson et al [5] and Hamada et al. [6] The latter technique involves beta prior distributions at the component level to incorporate test data of the form of k failures out of r demands to generate posterior component distributions. This allowed both upwards and downwards propagation of information but could not incorporate *overlapping data*. The Hamada et al method is written formally in equation (12).

$$\pi_1(\mathbf{p} | E) = \frac{L(E | \mathbf{p}) \pi_0(\mathbf{p})}{\int_{\forall \mathbf{p}'} L(E | \mathbf{p}') \pi_0(\mathbf{p}') d\mathbf{p}'} \quad \text{---(12)}$$

prior distribution: $\pi_0(\mathbf{p}) = \prod_{j=1}^n \left[(p_j)^{\alpha_j-1} (1-p_j)^{\beta_j-1} \right]$

likelihood function: $L(E | \mathbf{p}) = \left\{ \prod_{j=1}^n \left[(p_j)^{k_j} (1-p_j)^{r_j-k_j} \right] \right\} \left\{ \prod_{i=1}^m \left[(p_i^S)^{k_i^S} (1-p_i^S)^{r_i^S-k_i^S} \right] \right\}$

where \mathbf{p} is the set of n component failure probabilities $\{p_1, p_2, p_3, \dots, p_j, \dots, p_n\}$, p_i^S is the i^{th} system or sub-system failure probability expressed as a function of \mathbf{p} , m is the number of system and sub-system probabilities under consideration (i.e. the number of sensors), the prior beta distributions of component failure probabilities have parameters α and β , and evidence is shown in Table 2.

| Component/System | C_1 | C_2 | \dots | C_n | S_1 | S_2 | \dots | S_m |
|-----------------------------|-------|-------|---------|-------|---------|---------|---------|---------|
| Number of detected failures | k_1 | k_2 | \dots | k_n | k_1^S | k_2^S | \dots | k_m^S |
| Number of Demands | r_1 | r_2 | \dots | r_n | r_1^S | r_2^S | \dots | r_m^S |

Table 2: Expression of system level evidence for the Hamada et al method

The likelihood function becomes equation (13).

$$\pi_1(\mathbf{p} | E) = \frac{\left\{ \prod_{j=1}^n \left[(p_j)^{\alpha_j+k_j-1} (1-p_j)^{\beta_j+r_j-k_j-1} \right] \right\} \left\{ \prod_{i=1}^m \left[(p_i^{(S)})^{k_i^S} (1-p_i^{(S)})^{r_i^S-k_i^S} \right] \right\}}{\int_{\forall \mathbf{p}'} \left\{ \prod_{j=1}^n \left[(p_j')^{\alpha_j+k_j-1} (1-p_j')^{\beta_j+r_j-k_j-1} \right] \right\} \left\{ \prod_{i=1}^m \left[(p_i^{(S)'})^{k_i^S} (1-p_i^{(S)'})^{r_i^S-k_i^S} \right] \right\} d\mathbf{p}'} \quad \text{---(13)}$$

For the Hamada et al method, it can be seen that all evidence must be of the form of k detected failures out of r demands. An example of an application of the method of Hamada et al for a basic system is illustrated in Example 3.

The Hamada et al. method is computationally efficient to evaluate and should be used in scenarios involving *non-overlapping data* from binary-state on-demand systems. Graves et al. [7] generated a methodology that incorporates *overlapping data* for binary-state on-demand systems. It involves a four step algorithm primarily based on Boolean algebra and disjoint cut-set generation that satisfies overlapping multi-level data per demand.

Example 3: Non-overlapping data analysis of a binary-state on-demand system

Consider the basic two component on-demand series system in Example 1. The system level failure probability (or the probability of failure detection by sensor #1), p_1^S , is defined in terms of the failure probabilities of components 1 and 2, (p_1 and p_2 respectively). In this case, the set of component failure probabilities, \mathbf{p} , is $\{p_1, p_2\}$, noting that $n = 2$.

Test 2. *A series of 10 demands where 10 system level failures were detected by sensor #1 (sensor #2 was not involved).*

Firstly, assuming uniform prior distributions for the component failure probabilities and using the formalism in the Hamada et al approach, one can then summarize prior information in

through equations (14) and (15).

$$\alpha_1 = \beta_1 = 1 \quad \text{and} \quad \alpha_2 = \beta_2 = 1 \quad \text{---(14)(15)}$$

There is no component level evidence, and accordingly $k = r = 0$ for each component. The system level evidence can then be expressed as $k_1^S = 10$ and $r_1^S = 10$. There is only one set of higher level data (the system level), and therefore $m = 1$. Substitution into equation (13) yields:

$$\begin{aligned} \pi_1(\mathbf{p} | E) &= \frac{\prod_{j=1}^2 \left[(p_j)^{\alpha_j + k_j - 1} (1 - p_j)^{\beta_j + r_j - k_j - 1} \right] (p_1^S)^{k_1^S} (1 - p_1^S)^{r_1^S - k_1^S}}{\int_0^1 \int_0^1 \left\{ \prod_{j=1}^2 \left[(p_j')^{\alpha_j + k_j - 1} (1 - p_j')^{\beta_j + r_j - k_j - 1} \right] (p_1^S)^{k_1^S} (1 - p_1^S)^{r_1^S - k_1^S} \right\} dp_1' dp_2'} \\ &= \frac{(p_1)^{1+0-1} (1 - p_1)^{1+0-0-1} (p_2)^{1+0-1} (1 - p_2)^{1+0-0-1} (p_1^S)^{10} (1 - p_1^S)^{10-10}}{\int_0^1 \int_0^1 \left\{ (p_1')^{1+0-1} (1 - p_1')^{1+0-0-1} (p_2')^{1+0-1} (1 - p_2')^{1+0-0-1} (p_1^S)^{10} (1 - p_1^S)^{10-10} \right\} dp_1' dp_2'} \\ \pi_1(p_1, p_2 | \{k_1^S = 10, r_1^S = 10\}) &= \frac{(p_1 + p_2 - p_1 \times p_2)^{10}}{\int_0^1 \int_0^1 (p_1' + p_2' - p_1' \times p_2')^{10} dp_1' dp_2'} \quad \text{---(16)} \end{aligned}$$

The corresponding joint posterior distribution of p_1 and p_2 is illustrated below in Figure 4. It can be seen that from the data, the failure probabilities of p_1 and p_2 are likely to be high (i.e. close to 1). This makes ‘conceptual sense’, as it is known the system has a high failure probability and systemic failure occurs whenever component 1 or 2 fails.

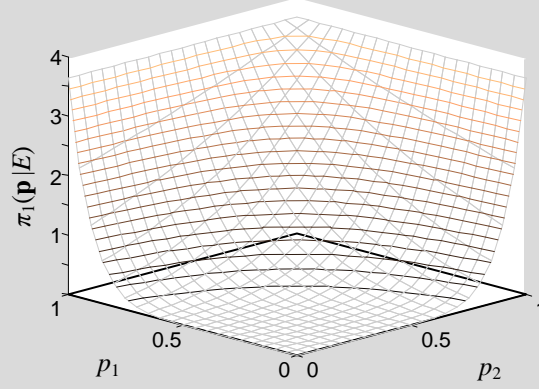


Figure 4: Joint posterior distribution of p_1 and p_2 - Test 2

(Returning to) Test 1. A series of 10 demands where 10 failures were detected at the system level and 1 failure was detected by sensor #2.

Data suggesting that component 1 has a low failure probability (1 failure from 10 demands) was gathered from sensor #2 concurrently with the systemic level data used in Test 2. The sensor level evidence can then be expressed as $k_1^S = 10$, $k_2^S = 1$ and $r_1^S = r_2^S = 10$. This suggests ‘conceptually’ that component 2 must have a very high failure probability to generate the high system level failure probability given that the data from sensor #2 suggests that component 1 is very reliable. The new posterior distribution is defined in equation (17).

$$\pi_1 \left(p_1, p_2 \left| \begin{cases} k_1^S = 10, r_1^S = 10 \\ k_2^S = 1, r_2^S = 10 \end{cases} \right. \right) = \frac{p_1 (1-p_1)^9 (p_1 + p_2 - p_1 \times p_2)^{10}}{\int_0^1 \int_0^1 p_1' (1-p_1')^9 (p_1' + p_2' - p_1' \times p_2')^{10} dp_1' dp_2'} \quad \text{---(17)}$$

The new joint posterior distribution of p_1 and p_2 is illustrated in Figure 5. The graph clearly suggests that p_1 is small (i.e. the failure probability of component 1 is low), and that p_2 is large

(i.e. the failure probability of component 2 is high).

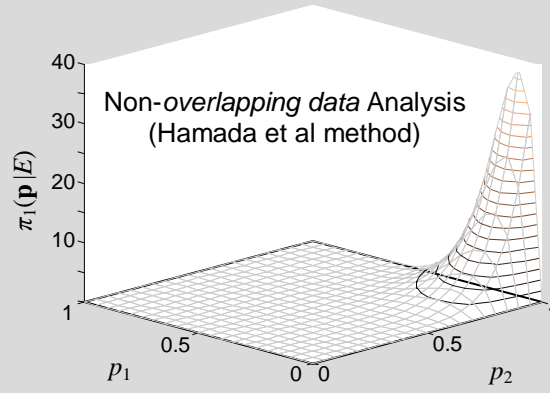


Figure 5: Joint posterior distribution of p_1 and p_2 -Test 1

2.2.2. Multi-state On-demand Systems

Multi-state systems are those where components exist in states that are classified by order of severity or degradation ranging from ‘functional’ to ‘failed’. Many systems exist where it is important to delineate between these states. Barlow and Wu [8] discuss the generalization of binary-state systems to multi-state systems. Graves et al [9] develop a fully Bayesian approach for incorporating multiple higher level *non-overlapping data* sets which is a generalization of their work in [6].

Each state of a multi-state system or component is represented by an integer from 0 to $(z - 1)$, where z is the total number of possible states. The state ‘0’ denotes total functionality while the state ‘ $z - 1$ ’ denotes total failure. Graves et al illustrate this in [9] by tabulating the possible states of five components in a system where there are four possible states in an example system.

| | Component | | | | |
|-----------------|-----------|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 |
| Possible States | 0 | 0 | 0 | 0 | 0 |
| | - | 1 | - | 1 | - |
| | 2 | 2 | - | 2 | - |
| | 3 | 3 | 3 | 3 | 3 |

Table 3: An example of possible states of components 1 to 5 of a given multi-state system

It can be seen in Table 3 that there are $z = 4$ possible states, but only some of the components can exist at all of them. The state ‘3’ (which is $z - 1$) denotes complete failure, and the state ‘0’ denotes complete functionality. Components 2 and 4 have a total of 4 possible states (including two degraded states that exist between complete functionality and complete failure), while components 3 and 5 have a total of 2 possible states.

The *state variable* for the j^{th} component is denoted x_j , and defines the state at which the variable exists. Equation (18) is a multi-state generalisation of the binary-state state variable equivalent introduced in equation (11).

$$x_j = \begin{cases} z - 1 & \text{if the } j^{\text{th}} \text{ component has completely failed} \\ \dots & \\ 1 & \text{if the } j^{\text{th}} \text{ component is in the first degraded state} \\ 0 & \text{if the } j^{\text{th}} \text{ component is completely functional} \end{cases} \quad \text{---(18)}$$

The states of all components in an entire system are defined by the *state vector* in equation (19).

$$\tilde{\mathbf{x}} = \{x_1, x_2, x_3, \dots, x_j, \dots, x_n\} \dots \text{for a system with } n \text{ components} \quad \text{---(19)}$$

Multi-state systems can be equally well represented graphically as binary-state systems through many methodologies including reliability block diagrams, fault trees and binary decision diagrams. These model representations are generally predicated on the rules that are represented graphically in Figure 6 for various gates associated with fault tree analysis.

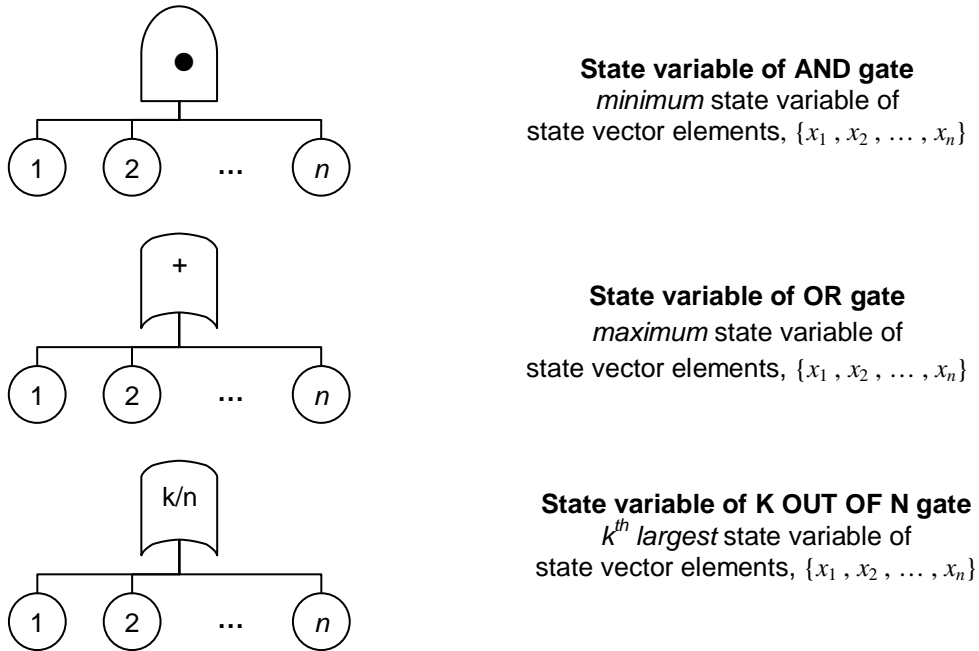


Figure 6: Multi-state gate state variable relationships

These basic rules form the core of multi-state system analysis and multiple tools exist for this purpose. Graves et al used the GROMIT algorithm [10] that generated mutually exclusive cut sets of component states for multi-state systems. This approach is illustrated in Example 4.

Example 4: Development of cut-sets for multi-state systems

Consider the basic three component system illustrated in Figure 7, which includes a ‘2 out of 3’ gate to define the state of the top event. The allowable states of the three components are also listed in a table that is included in Figure 7.

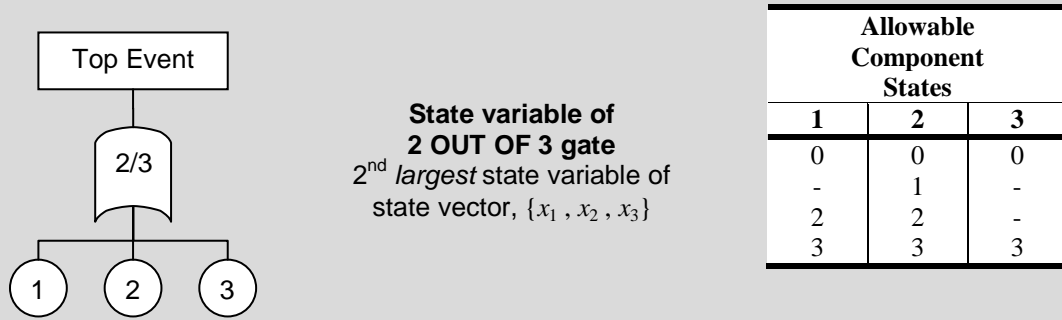


Figure 7: Basic three component system with a ‘2 out of 3’ gate defining the top event

The following mutually exclusive cut sets exist that define the top event:

$$x_{TE} = 0 \dots \{x_1=0, x_2=0\}; \{x_1=0, x_2=1, x_3=0\}; \{x_1=0, x_2=2, x_3=0\}; \{x_1=0, x_2=3, x_3=0\}; \{x_1=2, x_2=0, x_3=0\}; \{x_1=3, x_2=0, x_3=0\} \quad \text{---(20)}$$

$$x_{TE} = 1 \dots \{x_1=0, x_2=1, x_3=3\}; \{x_1=2, x_2=1, x_3=0\}; \{x_1=3, x_2=1, x_3=0\} \quad \text{---(21)}$$

$$x_{TE} = 2 \dots \{x_1=0, x_2=2, x_3=3\}; \{x_1=2, x_2=0, x_3=3\}; \{x_1=2, x_2=1, x_3=3\}; \{x_1=2, x_2=2\}; \{x_1=3, x_2=2, x_3=0\} \quad \text{---(22)}$$

$$x_{TE} = 3 \dots \{x_1=0, x_2=3, x_3=3\}; \{x_1=2, x_2=3, x_3=3\}; \{x_1=3, x_2=0, x_3=3\}; \{x_1=3, x_2=1, x_3=3\}; \{x_1=3, x_2=2, x_3=3\}; \{x_1=3, x_2=3\} \quad \text{---(23)}$$

Each cut set is simply a particular state vector or set of state vectors. Each state vector above implies a certain top-event state. For example, the state vector $\{x_1=0, x_2=1, x_3=3\}$ implies $x_{TE} = 1$. This is formally written as

$$\tilde{\mathbf{x}} = \{x_1 = 0, x_2 = 1, x_3 = 3\} \rightarrow x_{TE} = 1 \quad \text{---(24)}$$

Multi-state system analysis is not necessarily bound to the rules in Figure 6. For example, it may be desirable for the state of a gate to be described as the sum of the states of its subordinate components. In this instance, a basic two component system where the components exist in degraded states 1 and 2 respectively may imply that the system is considered to be in state 3 (the sum of states 1 and 2). To accommodate this, additional underlying rules and can be developed and mutually exclusive cut sets can be determined for each gate or system state.

The probability of each particular state vector occurring is:

$$\Pr(\tilde{\mathbf{x}} = \tilde{\mathbf{X}} | \mathbf{p}) = \prod_{j=1}^n p_j^{(x_j)} \quad \text{where } p_j^{(0)} = 1 - \sum_{x=1}^{z-1} p_j^{(x)} \quad \text{---(25)}$$

where the probability of the j^{th} component being in the x^{th} state is $p_j^{(x)}$, the set of $(z - 1)$ state probabilities for the j^{th} component is $\tilde{\mathbf{p}}_j = \{p_j^{(1)}, p_j^{(2)}, p_j^{(3)}, \dots, p_j^{(z-1)}\}$, the set of all n state probability sets for the system is $\mathbf{p} = \{\tilde{\mathbf{p}}_1, \tilde{\mathbf{p}}_2, \tilde{\mathbf{p}}_3, \dots, \tilde{\mathbf{p}}_j, \dots, \tilde{\mathbf{p}}_n\}$, and n is the number of components in the system.

The probability of a particular higher level state is given as:

$$\begin{aligned}
 p^{S(x^S)} &= \Pr(X^S = x^S) = \sum \Pr(\tilde{X} = \tilde{x}) \text{ for all } \tilde{x} \rightarrow x^S \\
 &= \sum_{x=0}^{z-1} \prod_{j=1}^n p_j^{(x)} \text{ where } p_j^{(0)} = 1 - \sum_{x=1}^{z-1} p_j^{(x)} \text{ for all } \tilde{x} \rightarrow x^S
 \end{aligned}
 \quad \text{---(26)}$$

The Bayesian methodology developed by Graves et al in [9] is a generalization of their previous methodology (that will be distinguished throughout this dissertation by being referred to as the Hamada et al method) in [6]. The Hamada et al method involves beta prior distributions of component failure probability and a likelihood function based on the binomial distribution (noting that this makes the prior distribution conjugate). The Graves et al method involves Dirichlet prior distributions of component state probabilities and a likelihood function based on the multinomial distribution (noting that this also makes the Dirichlet prior distribution a conjugate). The Dirichlet distribution (which is multi-variate) is defined in equation (27).

$$f(p^{(0)}, p^{(1)}, \dots, p^{(K)} | \alpha^{(0)}, \alpha^{(1)}, \dots, \alpha^{(K)}) = \frac{\Gamma\left(\sum_{x=0}^K \alpha^{(x)}\right)}{\prod_{x=0}^K \Gamma(\alpha^{(x)})} \left(\prod_{x=0}^K (p^{(x)})^{\alpha^{(x)}-1} \right) \quad \text{---(27)}$$

where $\{p^{(0)}, p^{(1)}, \dots, p^{(K)}\}$ are a set of non-negative random variables that satisfy $p^{(0)} + p^{(1)} + \dots + p^{(K)} \leq 1$ and $\{\alpha^{(0)}, \alpha^{(1)}, \dots, \alpha^{(K)}\}$ are a set of non-negative parameters.

In the Graves et al method, the random variables equate to component state probabilities. However, the sum of all state probabilities must sum to 1. This effectively removes one random

variable (as defining $K - 1$ random variables defines the remaining one if it must always be the case that it ‘completes’ the sum to 1). Accordingly, the Dirichlet distribution used in the Graves et al method is a particular case of the Dirichlet distribution in equation (27) where the number of random variables, K , becomes $z - 1$ and $p^{(0)} = 1 - (p^{(1)} + p^{(2)} + \dots + p^{(z-1)})$. It can be observed that by definition, the Beta distribution a particular case of the Dirichlet distribution described above where $z = 2$. The Graves et al method is then written formally as:

$$\pi_1(\mathbf{p} | E) = \frac{L(E | \mathbf{p}) \pi_0(\mathbf{p})}{\int_{\forall \mathbf{p}} L(E | \mathbf{p}') \pi_0(\mathbf{p}') d\mathbf{p}'} \quad \text{---(28)}$$

$$\begin{aligned} \text{prior distribution: } \pi_0(\mathbf{p}) &= \prod_{j=1}^n \pi_0(\tilde{p}_j) \\ &= \prod_{j=1}^n \left(\frac{\Gamma\left(\sum_{x=0}^{z-1} \alpha_j^{(x)}\right)}{\prod_{x=0}^{z-1} \Gamma(\alpha_j^{(x)})} \left(\prod_{x=1}^{z-1} (p_j^{(x)})^{\alpha_j^{(x)}-1} \right) \left(1 - \sum_{x=1}^{z-1} p_j^{(x)} \right)^{\alpha_j^{(0)}-1} \right) \\ &\propto \prod_{j=1}^n \left[\left(\prod_{x=1}^{z-1} (p_j^{(x)})^{\alpha_j^{(x)}-1} \right) \left(1 - \sum_{x=1}^{z-1} p_j^{(x)} \right)^{\alpha_j^{(0)}-1} \right] \end{aligned}$$

$$\text{likelihood function: } L(E | \mathbf{p}) = \prod_{i=1}^m \left(r_i^S ! \prod_{x=0}^{z-1} \frac{(p_i^{S(x)})^{k_i^{S(x)}}}{(k_i^{S(x)})!} \right) \propto \prod_{i=1}^m \left(\prod_{x=0}^{z-1} (p_i^{S(x)})^{k_i^{S(x)}} \right)$$

where the prior Dirichlet distributions of the j^{th} component state probabilities have parameters $\tilde{\alpha}_j = \{\alpha_j^{(0)}, \alpha_j^{(1)}, \alpha_j^{(2)}, \alpha_j^{(3)}, \dots, \alpha_j^{(z-1)}\}$ and evidence is input as shown in Table 4.

| System | State No | S_1 | S_2 | ... | S_m |
|----------------------------------|----------|----------------|----------------|-----|----------------|
| Number of observed system states | 0 | $k_1^{S(0)}$ | $k_2^{S(0)}$ | ... | $k_m^{S(0)}$ |
| | 1 | $k_1^{S(1)}$ | $k_2^{S(1)}$ | ... | $k_m^{S(1)}$ |
| | ... | ... | ... | ... | ... |
| | $z-1$ | $k_1^{S(z-1)}$ | $k_2^{S(z-1)}$ | ... | $k_m^{S(z-1)}$ |

Table 4: Expression of system level evidence for the Hamada et al method

For the Graves et al method, it can be seen that all evidence gathered at each sensor must be of the form of $\tilde{k}^S = \{k^{S(0)}, k^{S(2)}, k^{S(3)}, \dots, k^{S(z-1)}\}$ observed states. An example of an application of the method of Graves et al for a basic system is illustrated in Example 5.

Example 5: Non-overlapping data analysis of a multi-state on-demand system

Consider the basic two component, multi-state on-demand series system in Figure 8. The system level state probabilities (at the sensor #1 level), $\{p_1^{S(0)}, p_1^{S(1)}, p_1^{S(2)}, p_1^{S(3)}\}$ are defined in terms of the state probabilities of components 1 and 2, $\tilde{p}_1 = \{p_1^{(0)}, p_1^{(1)}, p_1^{(2)}, p_1^{(3)}\}$ and $\tilde{p}_2 = \{p_2^{(0)}, p_2^{(1)}, p_2^{(2)}, p_2^{(3)}\}$ respectively.

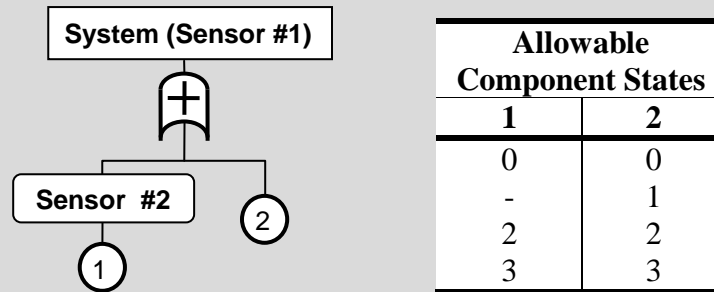


Figure 8: Multi-state two component series system

The mutually exclusive cut sets for each state detected by sensor #1 are:

$$x_1^S = 0 \dots \{x_1=0, x_2=0\} \quad \text{---(29)}$$

$$x_1^S = 1 \dots \{x_1=0, x_2=1\} \quad \text{---(30)}$$

$$x_1^S = 2 \dots \{x_1=0, x_2=2\} ; \{x_1=2, x_2=0\} ; \{x_1=2, x_2=1\} ; \{x_1=2, x_2=2\} \quad \text{---(31)}$$

$$x_1^S = 3 \dots \{x_1=0, x_2=3\} ; \{x_1=2, x_2=3\} ; \{x_1=3\} \quad \text{---(32)}$$

Similarly for each state detected by sensor #2:

$$x_2^S = 0 \dots \{x_1=0\} \quad \text{---(33)}$$

$$x_2^S = 2 \dots \{x_1=2\} \quad \text{---(34)}$$

$$x_2^S = 3 \dots \{x_1=3\} \quad \text{---(35)}$$

The system was subjected to a test of consisting of 10 separate demands with the gathered evidence shown in equations (36), (37) and (38).

$$\text{Sensor \#1:} \quad \tilde{\mathbf{k}}_1^S = \{k_1^{S(0)}, k_1^{S(1)}, k_1^{S(2)}, k_1^{S(3)}\} = \{2, 1, 4, 3\} \quad \text{---(36)}$$

$$\text{Sensor \#2:} \quad \tilde{\mathbf{k}}_2^S = \{k_2^{S(0)}, k_2^{S(1)}, k_2^{S(2)}, k_2^{S(3)}\} = \{3, 0, 4, 3\} \quad \text{---(37)}$$

$$\text{Evidence,} \quad E = \{\tilde{\mathbf{k}}_1^S, \tilde{\mathbf{k}}_2^S, \tilde{\mathbf{r}}^S\} \quad \text{---(38)}$$

The posterior distribution of the component state probabilities was evaluated using Markov-

Chain Monte Carlo (MCMC) simulation. The resultant marginal distributions of each state probability are illustrated in Figure 9.

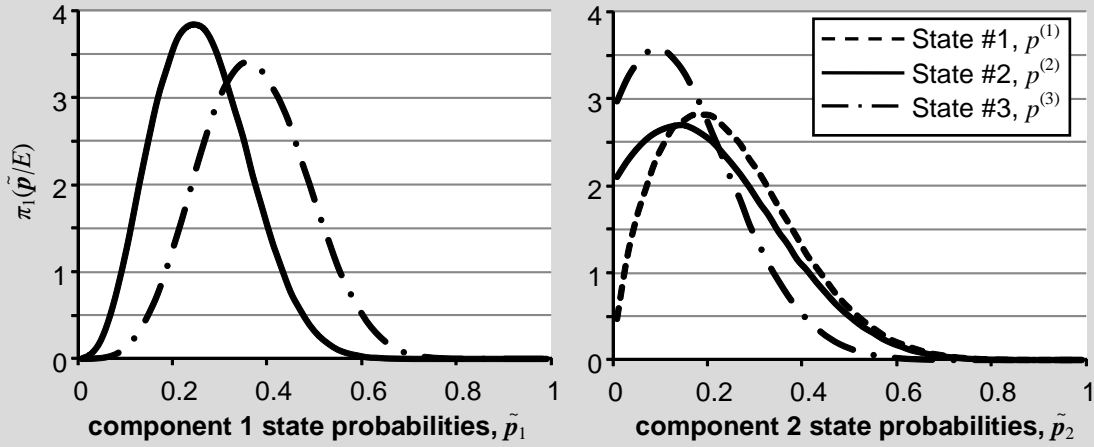


Figure 9: Marginal posterior distributions of each component state probability of system components in Figure 8 (using the Graves et al *non-overlapping* method).

As with the Hamada et al method for binary-state on-demand systems, the Graves et al method is the most computationally efficient method of conducting Bayesian analysis of *non-overlapping data* sets for multi-state on-demand systems. This dissertation includes a discussion on the differences between *overlapping* and *non-overlapping data*, and will later outline a methodology that can incorporate both *overlapping* and *non-overlapping data*.

2.3. AGGREGATION ERROR

Using higher level data to form posterior distributions of higher level events is *aggregate analysis*. The term ‘aggregate’ refers to the fact that system level reliability characteristics are

resultant of many substituent component failure probabilities and the analysis considers the combined effects of these characteristics holistically. Conversely, using lower level data to form posterior distributions of higher level reliability parameters is *disaggregate analysis* (i.e. component failure data analysed to generate system reliability parameter posterior distributions). Any difference between the two forms of analysis is called *aggregation error*. Aggregation error implies that the system is not correctly understood and therefore the relationships between higher level and component failure probabilities are misrepresented. It is important to note that even if the system is represented incorrectly, Bayesian analysis will improve higher level posterior distribution (but not those of the component parameters) [11]. Aggregation error has been studied in the field of reliability by Mosleh and Bier [12] and further developed by Azaiez and Bier [13] with Bayesian reliability estimates.

2.4. SENSOR PLACEMENT OPTIMISATION

Literature whose subject is ‘sensor placement optimisation’ by and large refers to sensors being placed at various locations within a physical process or mechanism as opposed to a system. These sensors are intended to detect faults or failure mechanisms as opposed to improving understanding of component and system reliability. In the field of *structural health monitoring* (and primarily aimed with identifying damage in the form of material and geometric changes of physical structures such as bridges) sensor optimisation is used to measure structural fidelity. [14] Sensors are also placed in physical networks. Watson, Greenberg and Hart discuss the problem of sensor optimisation to provide protection from terrorist attacks. [15] Dhillon and Chakrabarty discuss sensor placement optimisation for wireless networks. [16]

There is significant literature on system sensor placement that focuses on improving diagnostics. The point of difference of the sensor placement methodology developed in this dissertation is that information of the unknowns of interest (the component reliability characteristics, or state of knowledge) is the metric of optimisation.

2.5. SUMMARY

The majority of current reliability analysis methodologies that deal with data sets drawn from various levels, locations or sensors within a system only consider said sets as if they are *non-overlapping*. What this means is that (for example) if data was gathered simultaneously at the system and sub-system levels, traditional analysis limits inference based on an assumption that the data sets were gathered from distinctly separate test regimes. Chapter 2 has outlined many of the current methodologies that deal with *non-overlapping data*.

Overlapping data sets contain inherent dependencies that when analysed correctly, yield information that would otherwise be misinterpreted if they were otherwise analysed. In a simple example, system level data is contextualised by sub-system level data gathered at the same time, and understanding each in conjunction with the other infers more information about all relevant components. Methodologies are developed in the following chapters which allow *overlapping data* reliability analysis of a system to occur that illustrate its effect.

Chapter 3: Likelihood function of *overlapping data* of binary-state on-demand systems

3.1. INTRODUCTION

Graves et al. [7] proposed a method that incorporates *overlapping data* for binary-state on-demand systems. The methodology is based on a four step algorithm which relies on Boolean algebra and disjoint cut-set generation. As it considers each demand in isolation (i.e. sensor states for each demand must be known), the methodology cannot incorporate data that summarizes multiple demands on the system through a tallied number of states that each sensor detects. For the purpose of delineation, this methodology will be referred to as the *Overlapping Graves Method*. The methodology developed in this chapter allows *overlapping data* that is based on multiple demands of a binary-state on-demand system to be analysed in scenarios where the exact configuration of each *sensor information vector* (the states that each sensor detects) is not known for every specific demand.

3.2. LIKELIHOOD FUNCTION OF BINARY-STATE ON-DEMAND SYSTEMS

To develop the *overlapping data* likelihood function, permutations of possible instances of component failures that imply the observed evidence (and the probability of each permutation) need to be developed. The probability (or likelihood) of each permutation can then be substituted into equation (26). This is achieved in three steps listed in subsequent sub-headings.

3.2.1. Step 1: State Vector Analysis

All possible permutations of *state variables* for each component need to be analysed. Recall that the *state variable* for the j^{th} component of a binary-state on-demand system, x_j , is defined as:

$$x_j = \begin{cases} 0 & \text{if the } j^{\text{th}} \text{ component is functional} \\ 1 & \text{if the } j^{\text{th}} \text{ component has failed} \end{cases} \quad \text{---(11)}$$

The state of all components in the entire system can be defined by the *state vector*:

$$\tilde{\mathbf{x}} = \{x_1, x_2, x_3, \dots, x_j, \dots, x_n\} \dots \text{for a system with } n \text{ components} \quad \text{---(39)}$$

As there are multiple possible *state vectors*, the l^{th} *state vector* (and its constituent component *state variables*) is written as

$$\tilde{\mathbf{x}}_l = \{(x_1)_l, (x_2)_l, (x_3)_l, \dots, (x_j)_l, \dots, (x_n)_l\} \quad \text{---(40)}$$

3.2.2. Step 2: Structure Functions

Structure functions calculate the state of the system at sensor locations (i.e. at sub-system and system levels) based on the *state vector*. The *structure function* equates to 0 if the relevant sensor's (sub-) system is functional and 1 if it has failed.³

³ Many structure functions are used where '1' denotes functionality and '0' denotes failure. For ease of future calculation, the structure functions used in this dissertation are effectively reversed so that they generate system state variables.

$$\phi^S(\tilde{\mathbf{x}}) = \phi^S(\{x_1, x_2, x_3, \dots, x_j, \dots, x_n\}) = \begin{cases} 0 & \text{if the (sub-) system is functional} \\ 1 & \text{if the (sub-) system has failed} \end{cases} \quad \text{---(41)}$$

As each component is binary-state, there will be 2^n possible permutations and hence 2^n possible *state vectors*. The probability of each *state vector* occurring can be calculated based on individual component failure probabilities, p_j .

$$\Pr(\tilde{\mathbf{X}} = \tilde{\mathbf{x}} | \mathbf{p}) = \prod_{j=1}^n \left[(p_j)^{x_j} (1 - p_j)^{(1-x_j)} \right] \quad \text{---(42)}$$

3.2.3. Step 3: (Sub-) System Failure Combinations

The third step is to develop sets of combinations of r *state vectors* (recalling that r refers to the number of systemic demands in the data set) such that each combination generates the same number of detected failures as observed in the evidence set, $E = \{k_1^S, k_2^S, \dots, k_m^S, r\}$. It is important to note that the evidence set involves aggregates of failure detections for each sensor after r demands. Multiple evidence sets should not be combined into one aggregate set, as this represents a loss of information. For example, the evidence sets $E_1 = \{(k_1^S)_1, (k_2^S)_1, \dots, (k_m^S)_1, (r)_1\}$ and $E_2 = \{(k_1^S)_2, (k_2^S)_2, \dots, (k_m^S)_2, (r)_2\}$ should be separately analysed, and not combined to evidence set $E_{1+2} = \{(k_1^S)_1 + (k_1^S)_2, (k_2^S)_1 + (k_2^S)_2, \dots, (k_m^S)_1 + (k_m^S)_2, (r)_1 + (r)_2\}$.

$$\tilde{\mathbf{v}} = \{v_1, v_2, \dots, v_l, \dots, v_{2^n}\} \quad \text{---(43)}$$

where v_l is the number of occurrences of the l^{th} state vector, $\tilde{\mathbf{x}}_l$.

The a^{th} combination is defined by $\tilde{\mathbf{v}}_a = \left\{ (v_1)_a, (v_2)_a, \dots, (v_l)_a, \dots, (v_{2^n})_a \right\}$ ---(44)

where $(v_l)_a$ is the number of occurrences of the l^{th} state vector, $\tilde{\mathbf{x}}_l$, in the a^{th} combination.

The probability of occurrence of the a^{th} combination of *state vectors*, $\tilde{\mathbf{v}}_a$, given a set of lower level component failure probabilities is defined in equation (45).

$$\Pr(\tilde{\mathbf{v}}_a = \tilde{\mathbf{V}}_a | \mathbf{p}) = \Pr\left(\left\{ (v_1)_a = (V_1)_a, \dots, (v_l)_a = (V_l)_a, \dots, (v_{2^n})_a = (V_{2^n})_a \right\} | \mathbf{p}\right)$$

(which is simply a specific instance of the binomial distribution)

$$\begin{aligned} &= r! \prod_{l=1}^{2^n} \frac{\left(\Pr(\tilde{\mathbf{X}}_l = \tilde{\mathbf{x}}_l | \mathbf{p}) \right)^{(v_l)_a}}{(v_l)_a!} \quad \dots \text{substituting (42) yields:} \\ &= r! \prod_{l=1}^{2^n} \left\{ \frac{1}{(v_l)_a!} \left[\prod_{j=1}^n (p_j)^{(x_j)_l} (1-p_j)^{[1-(x_j)_l]} \right]^{(v_l)_a} \right\} \end{aligned} \quad \text{---(45)}$$

Since each *state vector* is linked with a specific demand, the total number of *state vectors* in the combination must sum to r :

$$\text{i.e.} \quad \sum_{l=1}^{2^n} v_l = r \quad \text{---(46)}$$

Each different combination implies a different number of failures detected by each sensor. The number of failures detected by the i^{th} sensor for the a^{th} combination of *state vectors*, $\tilde{\mathbf{v}}_a$, is defined in equation (47).

$$k_i^S = \sum_{l=1}^{2^n} (v_l)_a \phi_i^S(\tilde{\mathbf{x}}_l) \dots \text{ where } \tilde{\mathbf{v}}_a = \left\{ (v_1)_a, (v_2)_a, \dots, (v_l)_a, \dots, (v_{2^n})_a \right\} \quad \text{---(47)}$$

The matrix \mathbf{v}_E contains all combinations of state vectors that imply the evidence. If the a^{th} combination of *state vectors*, $\tilde{\mathbf{v}}_a$, implies the evidence set $E = \{k_1^S, k_2^S, \dots, k_m^S, r\}$ then it is an element of \mathbf{v}_E . For this to occur, the number of failures detected by each sensor implied by $\tilde{\mathbf{v}}_a$ must equal the number of detected failures in the evidence set, E .

$$\text{i.e. } \tilde{\mathbf{v}}_a \in \mathbf{v}_E \text{ iff } (k_i^S)_E = (k_i^S)_{\tilde{\mathbf{v}}_a} = \sum_{l=1}^{2^n} (v_l)_a \phi_i^S(\tilde{\mathbf{x}}_l) \dots \quad \forall i \in 1, 2, 3, \dots, m \quad \text{---(48)}$$

where $(k_i^S)_E$ is the number of failures detected by the i^{th} sensor in the evidence set E , and $(k_i^S)_{\tilde{\mathbf{v}}_a}$ is the number of implied failures detected by the i^{th} sensor for the a^{th} combination of state vectors, $\tilde{\mathbf{v}}_a$.

The *likelihood function* is the probability of observing the evidence, E , for a given instance of \mathbf{p} . The *likelihood function* is based on ascertaining the combinations of state vectors that imply the evidence, and then evaluating the probability of each combination is outlined in equation (49).

$\Pr(E | \mathbf{p}) = \sum \Pr(\text{combination of state vectors} | \mathbf{p}) \dots$ for all that imply the evidence

$$\begin{aligned}
 &= \sum_{\forall \tilde{\mathbf{v}}_a \in \mathbf{v}_E} \Pr(\tilde{\mathbf{V}}_a = \tilde{\mathbf{v}}_a | \mathbf{p}) \quad \dots \text{substituting (45) yields:} \\
 &= r! \sum_{\forall \tilde{\mathbf{v}}_a \in \mathbf{v}_E} \left[\prod_{l=1}^{2^n} \left\{ \frac{1}{(v_l)_a!} \left[\prod_{j=1}^n (p_j)^{(x_j)_l} (1-p_j)^{[1-(x_j)_l]} \right]^{(v_l)_a} \right\} \right] \quad \text{---(49)}
 \end{aligned}$$

Therefore, for a complex system with single occurrences of each component:

$$L(E | \mathbf{p}) \propto \sum_{\forall \tilde{\mathbf{v}}_a \in \mathbf{v}_E} \left[\prod_{l=1}^{2^n} \left\{ \frac{1}{(v_l)_a!} \left[\prod_{j=1}^n p_j^{(x_j)_l} (1-p_j)^{[1-(x_j)_l]} \right]^{(v_l)_a} \right\} \right] \quad \text{---(50)}$$

3.3. COMPARISON WITH NON-OVERLAPPING DATA METHODS – BINARY-STATE ON-DEMAND SYSTEMS

The Hamada et al method cannot incorporate information inherent in *overlapping data* sets as shown in Example 6, which compares to the *non-overlapping data* analysis in Example 3.

Example 6: *Overlapping data* analysis of binary-state on-demand systems

Consider the basic two component series system examined in Example 3 that was subjected to the first test regime:

(Recalling) Test 1. *A series of 10 demands where 10 failures were detected at the system level and 1 failure was detected by sensor #2.*

The structure functions for the system at each sensor location are:

$$\phi_1^S(\tilde{\mathbf{x}}) = \phi_1^S(\{x_1, x_2\}) = 1 - \prod_{j=1}^2 (1 - x_j) \quad \text{---(51)}$$

$$\phi_2^S(\tilde{\mathbf{x}}) = \phi_2^S(\{x_1, x_2\}) = x_1 \quad \text{---(52)}$$

Since there are two components (i.e. $n = 2$), the number of possible state vectors is $2^n = 2^2 = 4$.

The **state vectors** are listed in Table 5, along with the states detected by sensors that they imply and their inherent probabilities.

| State Vector # l | Component States | | State Vector $\tilde{\mathbf{x}}_l$ | States detected by sensors | | Probability $\Pr(\tilde{\mathbf{x}} = \tilde{\mathbf{X}}_l \mathbf{p})$ |
|--------------------------|---------------------|-----------|---|----------------------------------|----------------------------------|--|
| | $(x_1)_l$ | $(x_2)_l$ | | $\phi_1^S(\tilde{\mathbf{x}}_l)$ | $\phi_2^S(\tilde{\mathbf{x}}_l)$ | |
| 1 | 0 | 0 | {0,0} | 0 | 0 | $(1 - p_1)(1 - p_2)$ |
| 2 | 1 | 0 | {1,0} | 1 | 1 | $p_1(1 - p_2)$ |
| 3 | 0 | 1 | {0,1} | 1 | 0 | $(1 - p_1)p_2$ |
| 4 | 1 | 1 | {1,1} | 1 | 1 | p_1p_2 |

Table 5: State Vectors of system in **Figure 1.**

Since there are 10 demands (i.e. $r = 10$), there are 286 possible *state vector combinations*, $\tilde{\mathbf{v}}$, as each demand will invoke one of the 4 possible *state vectors*. A truncated list of combinations is listed in Table 6, along with the implied states detected by all sensors. There are only two possible combinations that imply the evidence.

| State Vector Combination No a | State Vector Combination, $\tilde{\mathbf{v}}_a$ | | | | Implied Evidence | | 1 if $\tilde{\mathbf{v}}_a \in \mathbf{v}_E$; 0 otherwise |
|--|--|-----------|-----------|-----------|--------------------------------------|--------------------------------------|---|
| | (no. of l^{th} state vectors) | | | | | | |
| | $(v_1)_a$ | $(v_2)_a$ | $(v_3)_a$ | $(v_4)_a$ | Failures detected by sensor #1 | Failures detected by sensor #2 | |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 9 | 1 | 0 | 0 | 1 | 1 | 0 |
| ... | | | | | | | |
| 65 | 0 | 1 | 9 | 0 | 10 | 1 | 1 |
| ... | | | | | | | |
| 121 | 0 | 0 | 9 | 1 | 10 | 1 | 1 |
| ... | | | | | | | |
| 285 | 0 | 0 | 1 | 9 | 10 | 9 | 0 |
| 286 | 0 | 0 | 0 | 10 | 10 | 10 | 0 |

Table 6: Possible state vector combinations of system in Figure 1.

The normalized likelihood function given by equation (50) is substituted into equation (28) with uniform prior distributions to generate the posterior distribution plotted in Figure 10, along with the posterior distribution derived by the Hamada et al method. The normalized percentage error of the Hamada et al method is illustrated in Figure 11.

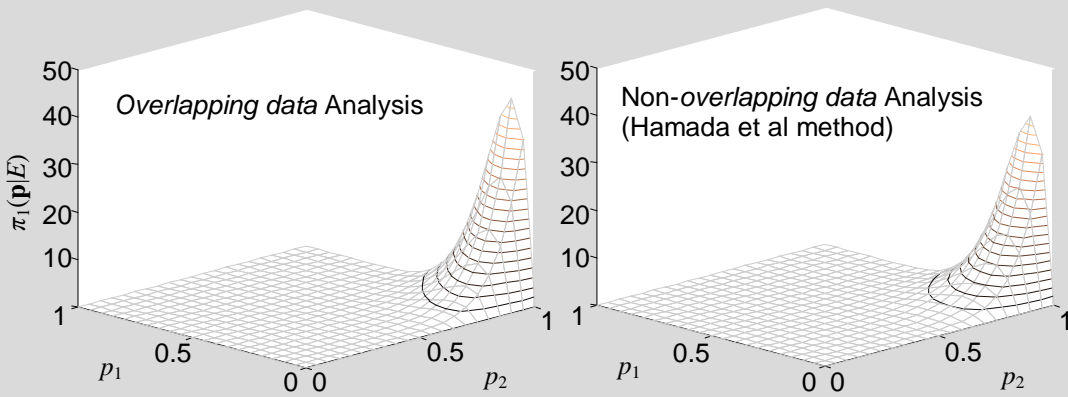


Figure 10: Joint posterior distributions of p_1 and p_2 from Test 1

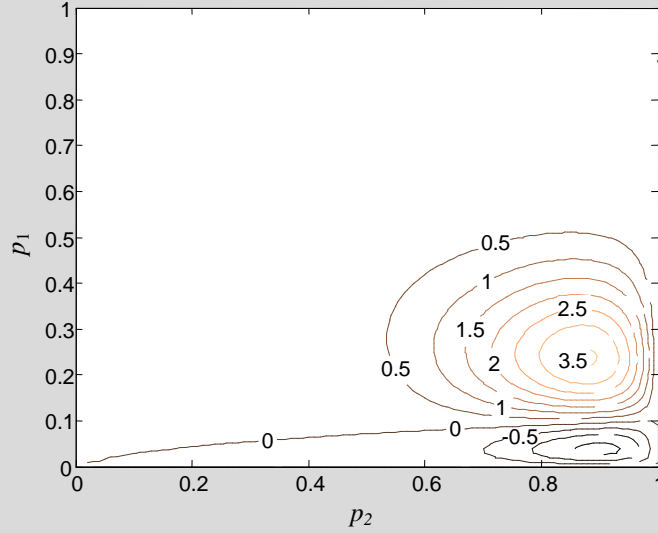


Figure 11: Percentage error in *non-overlapping data* likelihood function

(maximum normalised error = 3.55 %)

The *overlapping* posterior distribution illustrated in Figure 10 can fortuitously be replicated in this instance by the *Overlapping Graves et al. Method*. The evidence (10 demands, 10 systemic failures, 1 failure detected by sensor #2) is equivalent to one demand where the system fails **and** sensor #2 detects failure, and 9 demands where the system fails **and** sensor #2 does not detect failure. As failure detection for each demand is thus known, the *Overlapping Graves et al. Method* can be applied.

The two posterior distributions in Figure 10 are approximately equal, with the normalized Hamada et al (*non-overlapping*) posterior distribution differing by less than 3.55 % from the normalised *overlapping data* posterior distribution. This good approximation is due to the nature of the evidence, E . The inference of the system level data contains a relatively small ‘observational’ dependence on the inference of the sensor #2 data. Example 7 illustrates the

potential effect for an evidence set that contains a significant dependence between the data levels.

Example 7: Non-overlapping and overlapping data analysis of an on-demand system.

Consider the basic two component series system examined in Example 3 and Example 6 subjected to a third test regime:

***Test 3.** A series of 10 demands where 5 system level failures were detected by sensor #1 and 5 failures were detected by sensor #2.*

The evidence can be interpreted in two separate ways by considering it to be either *overlapping* or *non-overlapping*:

Overlapping data. The implication of evidence set E is illustrated in Figure 12. In effect, component 1 has been observed to fail 5 times on 10 demands and component 2 has been observed to fail 0 times on 5 demands.

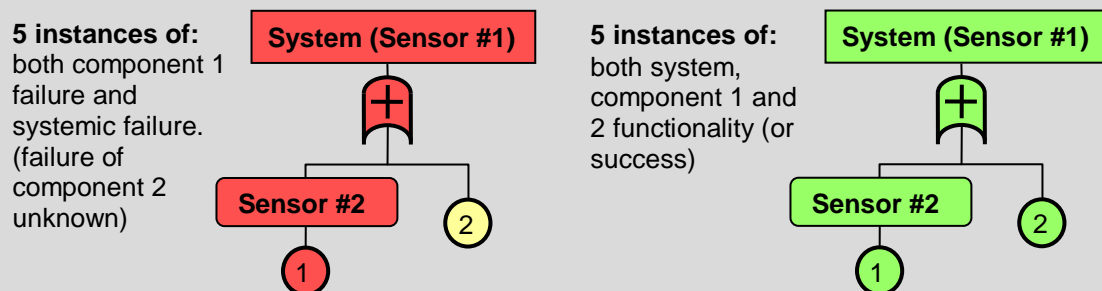


Figure 12: Overlapping data Evidence

Non-overlapping data. The implication of evidence set E is illustrated in Figure 13. In effect, component 1 has been observed to fail 5 times on 10 demands whilst the system has separately been observed to fail another 5 times on 10 demands. This introduces additional uncertainty on both components, and will imply that component 2 has a higher likelihood of having a high failure probability.

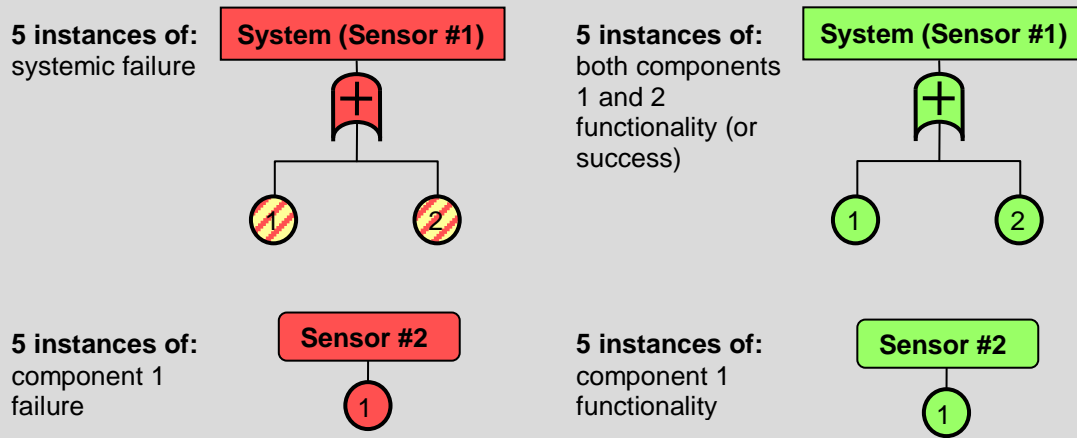


Figure 13: Non-overlapping data Evidence

The normalized likelihood function given by equation (50) is substituted into equation (28) with uniform prior distributions to generate a posterior distribution plotted in Figure 14, along with the posterior distribution derived by the Hamada et al method. The normalised error associated with treating *overlapping data* as *non-overlapping* (i.e. the difference between the two graphs in Figure 14) are illustrated in Figure 15.

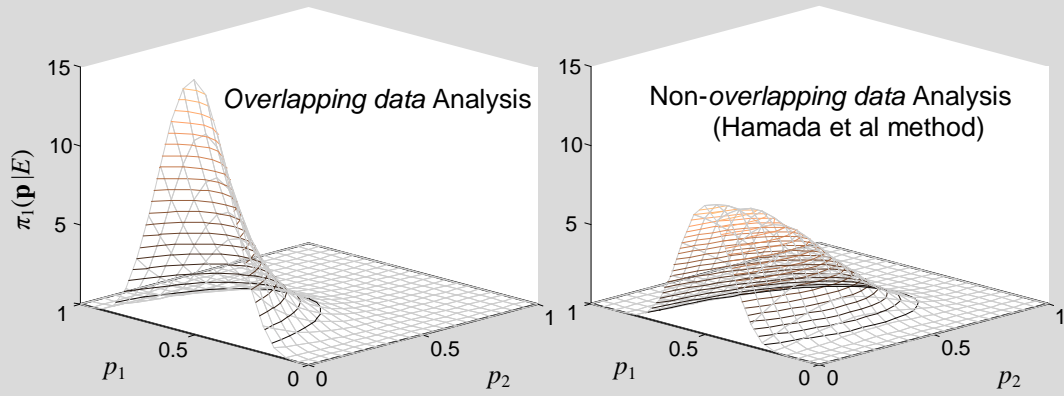


Figure 14: Joint posterior distributions of p_1 and p_2 from test 3 evidence

The graph generated by the *non-overlapping data* analysis in the right of Figure 14 is more ‘diffuse’ or less concentrated on a single set. This means that the *overlapping data* posterior distribution is significantly less uncertain, and hence contains more information.

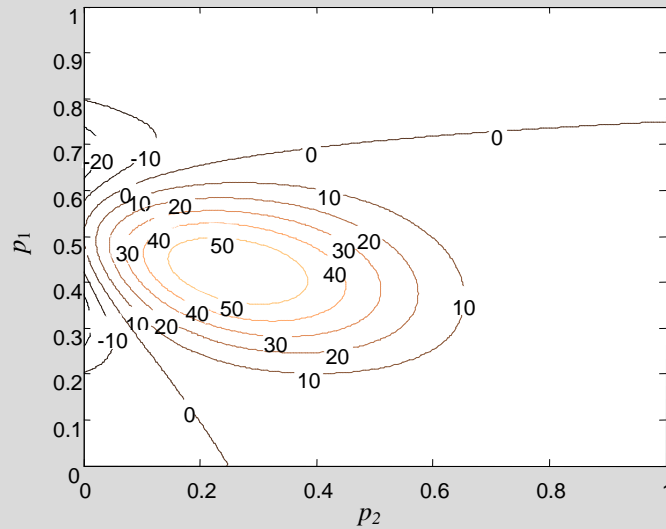


Figure 15: Percentage error in non-overlapping data likelihood function

(maximum error = 59.97 %)

Treating *overlapping data* as *non-overlapping* can also incorrectly imply non-existent information. Higher level data may become effectively redundant in the context of lower level data, which may mask information about some components. Treating the data as non-overlapping may incorrectly imply information about these components in such instances.

Example 8: Non-overlapping and *overlapping data* analysis of a binary-state on-demand system.

Consider the basic two component series system examined in previous examples that is subjected to a fourth test regime:

***Test 4.** A series of 10 demands where 10 system level failures were detected by sensor #1 and 10 failures were detected by sensor #2.*

The evidence can be interpreted in two separate ways by considering it to be either *overlapping* or *non-overlapping*:

Overlapping data. The implication of evidence set E is illustrated in Figure 16. In effect, component 1 has been observed to fail 10 times on 10 demands (as every time sensor #2 detects failure, sensor #1 detects failure) and component 2 has not been observed at all.

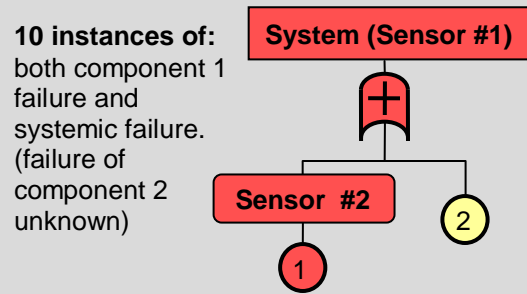


Figure 16: *Overlapping data Evidence*

Non-overlapping data. The implication of evidence set E is illustrated in Figure 17. In effect, component 1 has been observed to fail 10 times on 10 demands, whilst the system has separately been observed to fail 10 times on 10 demands.

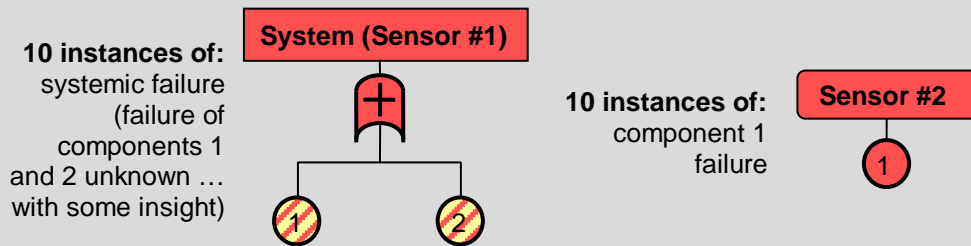


Figure 17: *Non-overlapping data Evidence*

The normalized likelihood function given by equation (50) is substituted into equation (28) with uniform prior distributions to generate a posterior distribution plotted in Figure 18, along with the posterior distribution derived by the Hamada et al method. The normalized percentage error of the Hamada et al method is illustrated in Figure 19.

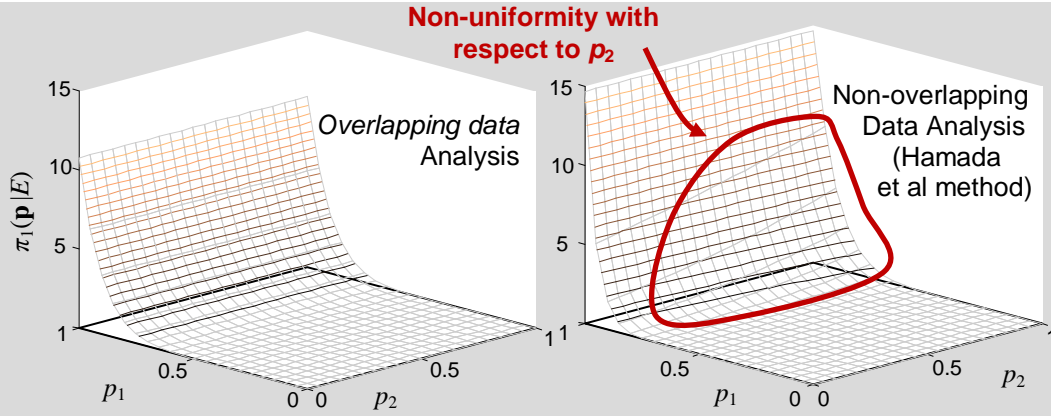


Figure 18: Joint posterior distributions of p_1 and p_2 from test 4 evidence

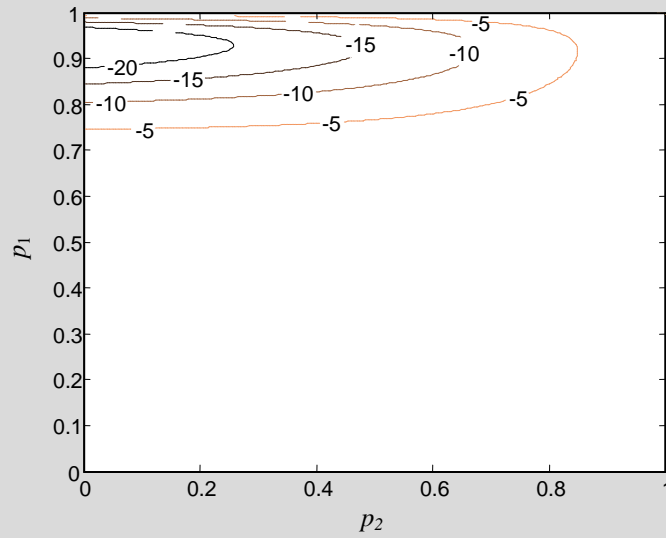


Figure 19: Percentage error in non-overlapping data likelihood function

(maximum error = 24.99 %)

Since all demands generated failures that were detected at the system and sensor # 2 levels, the behaviour of component 2 is effectively hidden. As it is a series system, the success or failure of component 2 does not alter the data/evidence set. Accordingly, there is no information about component 2, and the joint likelihood function has to be uniform with respect to the failure

probability of component 2. The non-overlapping posterior distribution in Figure 18 suggests that a higher failure probability in component 2 is more likely. This information is has no base, and represents another inaccuracy when *overlapping data* is analysed as if it were *non-overlapping*.

When there is only one data set (as in data is gathered from only one sensor), the data cannot be considered to be overlapping or non-overlapping. In this case, the Hamada et al and *overlapping data* methods generate identical likelihood functions.

Example 9: Non-overlapping and *overlapping data* analysis of a binary-state on-demand system.

Consider the basic two component series system examined in previous examples subjected to the second test regime:

(Recalling) Test 2. *A series of 10 demands where 10 system level failures were detected by sensor #1 (sensor #2 was not involved).*

The implication of evidence set E is illustrated in Figure 20. Whether the data is treated as overlapping or non-overlapping as it is irrelevant in the case of only one data set.

10 instances of:
systemic failure
(failure of
components 1
and 2 unknown ...
with some insight)

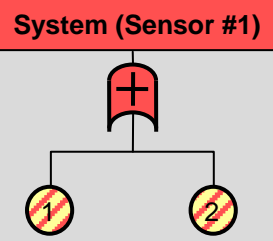


Figure 20: Data Inference

The normalized likelihood function given by equation (50) is plotted in Figure 21 along with the likelihood function derived by the Hamada et al method.

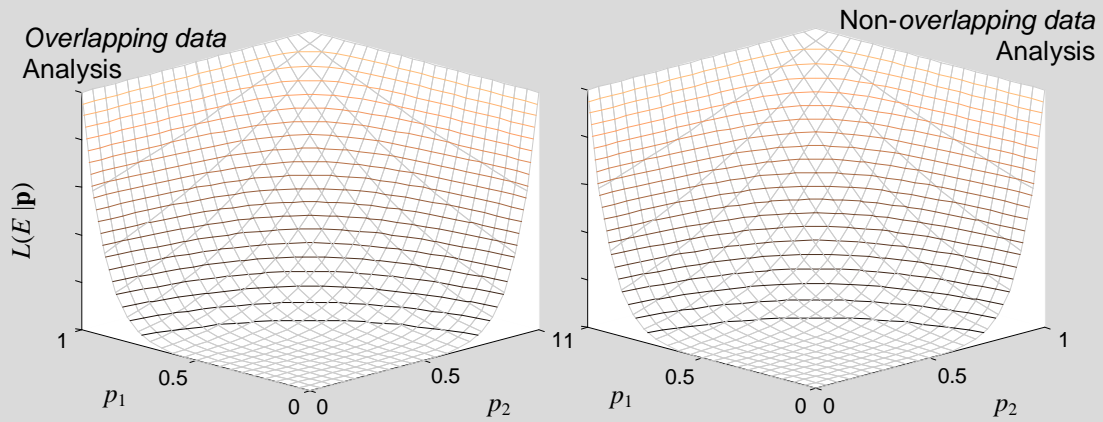


Figure 21: Normalized likelihood functions of p_1 and p_2 from test 1 evidence (identical)

The normalized percentage discrepancy between the two likelihood functions is due only to calculation errors that do not exceed $2.22 \times 10^{-13} \%$.

3.4. MULTIPLE INSTANCES OF IDENTICAL COMPONENTS

Apostolakis and Kaplan outline the nature of *state of knowledge* dependence through multiple instances of identical components. [17] The underlying failure probability of each identical component (i.e. components of the same *component type*) is the same, and modifies the likelihood function accordingly. The unknown of interest, \mathbf{p} , then becomes the set of *component type failure probabilities* where n is the number of *component types*. The total number of components becomes n' (and therefore $n \leq n'$). The *component types* are numbered $1, 2, 3, \dots, j, \dots, n$, and the *components* are numbered $1, 2, 3, \dots, b, \dots, n'$. The *component type* of the b^{th} component is denoted j_b . The likelihood function for a binary-state on-demand system is modified from equation (50) when there are multiple occurrences of the same component to equation (53).

$$L(E | \mathbf{p}) \propto \sum_{\forall \tilde{\mathbf{v}}_a \in \mathbf{V}_E} \left[\prod_{l=1}^{2^{n'}} \left\{ \frac{1}{(v_l)_a!} \left[\prod_{b=1}^{n'} (p_{j_b})^{(x_b)_l} (1 - p_{j_b})^{[1 - (x_b)_l]} \right]^{(v_l)_a} \right\} \right] \quad \text{---(53)}$$

where the unknown of interest, $\mathbf{p} = \{p_1, p_2, \dots, p_j, \dots, p_n\}$ is the set of n lower level *component type* failure probabilities, p_{j_b} is the failure probability of the b^{th} component (which is the failure probability of j_b^{th} *component type*), $\tilde{\mathbf{v}}_a$ is the a^{th} combination of r *state vectors* (each *state vector* comprises of n' component states), $(v_l)_a$ is the number of occurrences of the l^{th} *state vector* in $\tilde{\mathbf{v}}_a$ and $(x_b)_l$ is the *state variable* of the b^{th} component in the l^{th} *state vector*.

Example 10: *Overlapping data* analysis of a binary-state on-demand system with multiple instances of the same component

Consider a basic two component series system similar to that examined in previous examples. It is made up of two identical components of type A, and is illustrated in Figure 22.

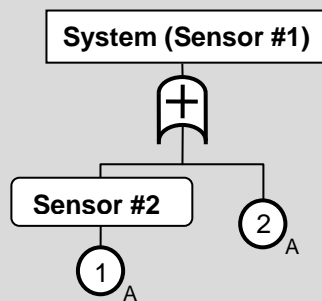


Figure 22: Basic two identical component series system

It is subjected to the following test:

Test 5. A series of 10 demands where 3 system level failures were detected by sensor #1 and no failures were detected by sensor #2.

As the data is *overlapping*, the implication of evidence set E is illustrated in Figure 23. In effect, component 1 has been observed to fail 0 times on 10 demands, and component 2 has been observed to fail 3 times on 10 demands.

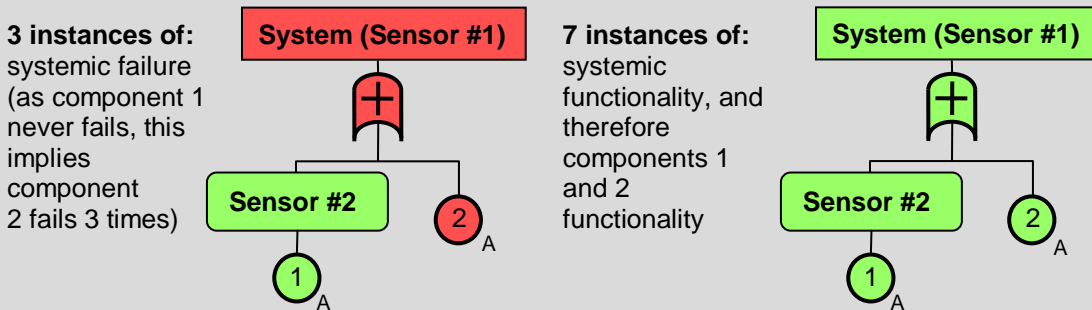


Figure 23: *Overlapping data Evidence*

As components 1 and 2 are identical (i.e. the same component type), the evidence is equivalent to a single component of type 1 failing 3 times on 20 demands. When considering the equivalent evidence set $E = \{k = 3 ; r = 20\}$, the likelihood function is:

$$L(E = \{k = 3; r = 20\} | p_1) = p_1^k (1 - p_1)^{r-k} = p_1^3 (1 - p_1)^{17} \quad \text{---(54)}$$

The normalized likelihood function given by equation (53) is plotted in Figure 24, along with the likelihood function derived in equation (54). The normalized percentage discrepancy between the two methods below is due to computational errors and does not exceed $3.33 \times 10^{-14} \%$.

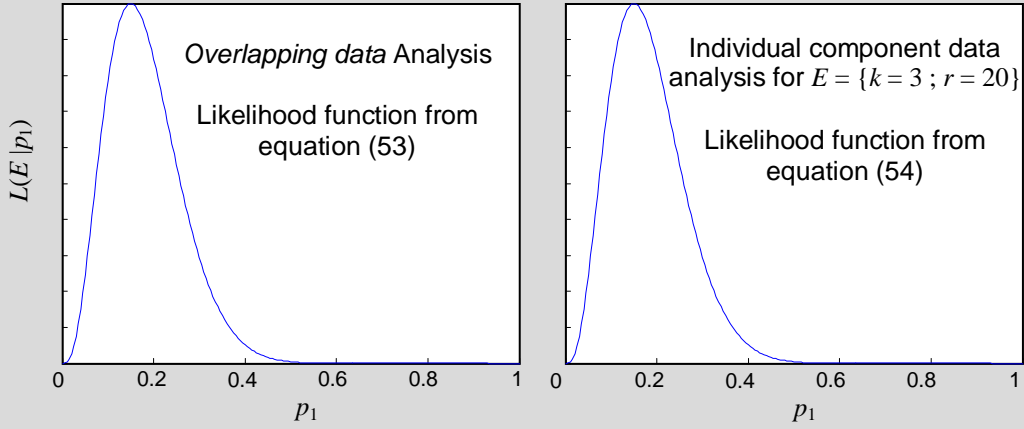


Figure 24: Identical Normalized likelihood functions of p_1 from test 5 evidence

3.5. LIMITATIONS ON PREVIOUS METHODOLOGIES

As discussed, the majority of existing methodologies can only deal with *non-overlapping data*. The only other methodology that deals with *overlapping data*, the *Overlapping Graves et al. Method*, can only deal with evidence where the sensor data is precisely known for each demand and is not aggregated. The method proposed in this chapter can deal with not only this form of evidence, but evidence in the form of multiple demands where aggregate numbers of failures are known for each sensor.

Example 11: Limitation of the *Overlapping Graves et al. method*

Consider the 4 component, 2 sensor system in Figure 25 that is subjected to a sixth test regime:

Test 6. A series of 10 demands where 10 system level failures were detected by sensor #1 and 10 failures were detected by sensor #2.

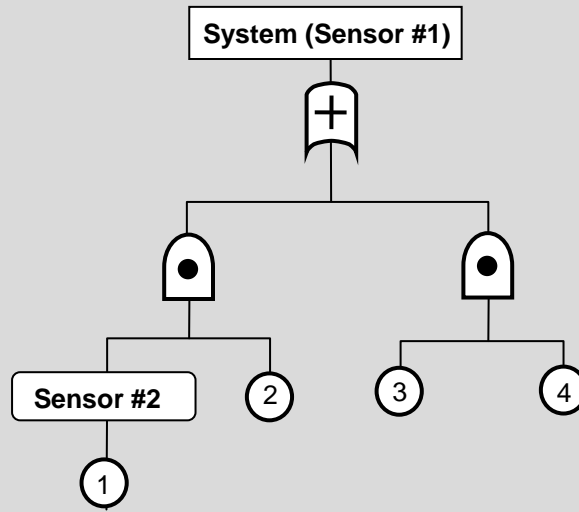


Figure 25: Three sensor on-demand system with *paths of apparent influence* indicated

In this example, it is not known if all of the demands where sensor #2 detected failure coincide with system level failure, if all of the demands where sensor #2 detected failure coincided with system level success, or any permutation in between. The *Overlapping Graves et al. Method* requires the precise permutation of sensor data to be known for each demand, and hence cannot incorporate evidence or data of this nature.

3.6. SUMMARY

Whilst fully Bayesian methodologies have been developed to incorporate data at various levels within binary-state on-demand systems, the majority have been constrained to treat all data as *non-overlapping*. This ignores the dependencies between the *overlapping data* sets and effectively removes or misinterprets inherent information. An *overlapping data* likelihood function for binary-state on-demand systems was developed in this chapter that incorporates

these inherent dependencies and generate the correct inference through Bayes' Theorem. Several examples were developed to highlight the effect of the additional information *overlapping data* sets contain and how it can be used to correctly improve the state of knowledge (which in the context of binary-state on-demand systems is the set of *component type* failure probabilities). These examples included simple systems for the sake of illustration, but the methodology is equally applicable to complex systems. The flexibility of the likelihood function was also developed further to incorporate multiple instances of identical components. Through state of knowledge dependence, the resultant *overlapping data* Bayesian method completely incorporates all information and evidence that can possibly be generated or observed in complex, binary-state on-demand systems.

Chapter 4: Likelihood function of *overlapping data* of multi-state on-demand systems

4.1. INTRODUCTION

Chapter 3 outlined a methodology for Bayesian analysis of *overlapping data* from binary-state on-demand systems. A methodology is outlined in this chapter that generalises the likelihood function developed in the previous chapter for multi-state on-demand systems.

4.2. LIKELIHOOD FUNCTION OF MULTI-STATE ON-DEMAND SYSTEMS

The developmental steps for the *overlapping data* likelihood function of multi-state on-demand systems are identical to that for binary-state systems in that permutations of possible component states (and the probability of each permutation) need to be developed. The probability (or likelihood) of observing any one of the permutations that imply the observed evidence can then be substituted into equation (28). This is achieved using the three steps developed in Chapter 3.

4.2.1. Step 1: State Vector Analysis

All possible permutations of *state variables* for each component need to be analysed. The system *state vector* is defined in equation (39):

$$\tilde{\mathbf{x}} = \{x_1, x_2, x_3, \dots, x_j, \dots, x_n\} \dots \text{for a system with } n \text{ components} \quad \text{---(39)}$$

where x_j is the state variable of the j^{th} component.

As there are multiple possible *state vectors*, the l^{th} *state vector* (and its constituent component *state variables*) is written as:

$$\tilde{\mathbf{x}}_l = \{(x_1)_l, (x_2)_l, (x_3)_l, \dots, (x_j)_l, \dots, (x_n)_l\} \quad \text{---(40)}$$

4.2.2. Step 2: Structure Functions

Structure functions calculate the state of (sub-) system levels where sensors are located based on the *state vector* and is dependent on the system logic. The *structure function* returns the state of a higher level gate or sensor, and equates to $(z - 1)$ if the relevant sensor's (sub-) system has completely failed and 0 if it is fully functional.

$$x_i^S = \phi_i^S(\tilde{\mathbf{x}}) = \phi_i^S(\{x_1, x_2, x_3, \dots, x_j, \dots, x_n\})$$

$$= \begin{cases} z - 1 & \text{if the (sub-) system has completely failed} \\ \dots & \dots \\ 1 & \text{if the (sub-) system is in the first degraded state} \\ 0 & \text{if the (sub-) system is completely functional} \end{cases} \quad \text{---(55)}$$

which is a generalisation of the binary-state equivalent developed in equation (41).

There will be z^n permutations and hence z^n possible *state vectors*. The probability of each *state vector* occurring can be calculated based on individual component state probabilities. The probability of the j^{th} component being in the x^{th} state is denoted $p_j^{(x)}$. The probability of the state vector $\tilde{\mathbf{x}}$ occurring is outlined in equation (25).

$$\Pr(\tilde{\mathbf{x}} = \tilde{\mathbf{X}} | \mathbf{p}) = \prod_{j=1}^n p_j^{(x_j)} \text{ where } p_j^{(0)} = 1 - \sum_{x=1}^{z-1} p_j^{(x)} \quad \text{---(25)}$$

4.2.3. Step 3: (Sub-) System Failure Combinations

The third step is to develop multiple combinations of r *state vectors* (recalling that r refers to the number of systemic demands in the data set) such that each combination generates the same number of observed states as the evidence set, $E = \{\tilde{\mathbf{k}}_1^S, \tilde{\mathbf{k}}_2^S, \dots, \tilde{\mathbf{k}}_m^S, r\}$, recalling that each vector $\tilde{\mathbf{k}}_i^S = \{k_i^{S(0)}, k_i^{S(1)}, \dots, k_i^{S(z-1)}\}$ defines the number of states detected by the i^{th} sensor. The inclusion of the number of demands, r , as a separate value in the evidence set, E , is technically redundant as the number of states that each sensor detects, expressed by $\tilde{\mathbf{k}}^S$, sums to r for all sensors.

$$\tilde{\mathbf{v}} = \{v_1, v_2, \dots, v_l, \dots, v_{z^n}\} \quad \text{---(56)}$$

where v_l is the number of occurrences of the l^{th} state vector, $\tilde{\mathbf{x}}_l$... which is a generalisation of the binary-state equivalent in equation (43).

$$\text{The } a^{\text{th}} \text{ combination is defined by: } \tilde{\mathbf{v}}_a = \{(v_1)_a, (v_2)_a, \dots, (v_l)_a, \dots, (v_{z^n})_a\} \quad \text{---(57)}$$

where $(v_l)_a$ is the number of occurrences of the l^{th} state vector in the a^{th} combination ... which is a generalisation of the binary-state equivalent in equation (44).

The probability of a particular combination of *state vectors*, $\tilde{\mathbf{v}}_a$, occurring given a set of lower level component failure probabilities is listed in equation (58).

$$\Pr(\tilde{\mathbf{V}}_a = \tilde{\mathbf{v}}_a | \mathbf{p}) = \Pr\left(\left\{(V_1)_a = (v_1)_a, \dots, (V_l)_a = (v_l)_a, \dots, (V_{2^n})_a = (v_{2^n})_a\right\} | \mathbf{p}\right)$$

(which is simply a specific instance of the multinomial distribution)

$$\begin{aligned} &= r! \prod_{l=1}^{z^n} \frac{\left(\Pr(\tilde{\mathbf{X}}_l = \tilde{\mathbf{x}}_l | \mathbf{p})\right)^{(v_l)_a}}{(v_l)_a!} \quad \dots \text{ substituting (25) yields:} \\ &= r! \prod_{l=1}^{z^n} \left\{ \frac{1}{(v_l)_a!} \left(\prod_{j=1}^n p_j^{(x_j)} \right)^{(v_l)_a} \right\} \end{aligned} \quad \text{---(58)}$$

which is a generalisation of the binary-state equivalent in equation (45) that utilised the binomial distribution.

Since each *state vector* is linked with a specific demand, the total number of *state vectors* in the combination must sum to r :

$$\sum_{l=1}^{z^n} v_l = r \quad \text{---(59)}$$

which is a generalisation of the binary-state equivalent in equation (46).

Each different combination implies a different number of states observed by each sensor. The number of x^{th} states observed by the i^{th} sensor for the a^{th} combination of *state vectors*, $\tilde{\mathbf{v}}_a$, is defined in equation (60).

$$k_i^{S(x)} = \sum_{l=1}^{z^n} (v_l)_a \begin{cases} 1 & \text{if } \phi_i^S(\tilde{\mathbf{x}}_l) = x \\ 0 & \text{if } \phi_i^S(\tilde{\mathbf{x}}_l) \neq x \end{cases} \quad \dots \text{ where } \tilde{\mathbf{v}}_a = \left\{ (v_1)_a, (v_2)_a, \dots, (v_l)_a, \dots, (v_{z^n})_a \right\} \quad \text{---(60)}$$

The matrix \mathbf{v}_E contains all combinations of state vectors that imply the evidence. If the a^{th} combination of *state vectors*, $\tilde{\mathbf{v}}_a$, implies the evidence set $E = \{\tilde{\mathbf{k}}_1^S, \tilde{\mathbf{k}}_2^S, \dots, \tilde{\mathbf{k}}_m^S, r\}$, then $\tilde{\mathbf{v}}_a$ is an element of \mathbf{v}_E . For this to occur, the number of states detected by each sensor implied by $\tilde{\mathbf{v}}_a$ must equal the number contained in the evidence set.

$$\text{i.e. } \tilde{\mathbf{v}}_a \in \mathbf{v}_E \text{ iff } (\tilde{\mathbf{k}}_i^S)_E = (\tilde{\mathbf{k}}_i^S)_{\tilde{\mathbf{v}}_a} \quad \dots \quad \forall i \in 1, 2, 3, \dots, m \quad \text{---(61)}$$

where $(\tilde{\mathbf{k}}_i^S)_E$ is the vector of the number of states detected by the i^{th} sensor in the evidence set E , and $(\tilde{\mathbf{k}}_i^S)_{\tilde{\mathbf{v}}_a}$ is the vector of the number of implied states detected by the i^{th} sensor for the a^{th} combination of state vectors, $\tilde{\mathbf{v}}_a$, each element of $(\tilde{\mathbf{k}}_i^S)_{\tilde{\mathbf{v}}_a}$ being defined by equation (60) ... all of which is a generalisation of the binary-state equivalent in equation (48).

The *likelihood function* is the probability of observing the evidence, $E = \{\tilde{\mathbf{k}}_1, \tilde{\mathbf{k}}_2, \dots, \tilde{\mathbf{k}}_m, r\}$ for a given instance of \mathbf{p} . The *likelihood function* is based on ascertaining the combinations of state vectors that imply the evidence, and then evaluating the probability of each combination:

$$\Pr(E | \mathbf{p}) = \sum \Pr(\text{combination of state vectors} | \mathbf{p}) \dots \text{for all that imply the evidence}$$

$$= \sum_{\forall \tilde{\mathbf{v}}_a \in \mathbf{v}_E} \Pr(\tilde{\mathbf{V}}_a = \tilde{\mathbf{v}}_a | \mathbf{p}) \quad \dots \text{substituting (58) yields:}$$

$$= r! \sum_{\forall \tilde{\mathbf{v}}_a \in \mathbf{v}_E} \left[\prod_{l=1}^{z^n} \left\{ \frac{1}{(v_l)_a!} \left(\prod_{j=1}^n p_j^{((x_j)_l)} \right)^{(v_l)_a} \right\} \right] \text{ where } p_j^{(0)} = 1 - \sum_{x=1}^{z-1} p_j^{(x)} \quad \text{---(62)}$$

Therefore, for a complex system with single occurrences of each component:

$$L(E | \mathbf{p}) = \Pr(\tilde{\mathbf{x}} \rightarrow E | \mathbf{p}) \propto \sum_{\forall \tilde{\mathbf{v}}_a \in \mathbf{v}_E} \left[\prod_{l=1}^{z^n} \left\{ \frac{1}{(v_l)_a!} \left(\prod_{j=1}^n p_j^{((x_j)_l)} \right)^{(v_l)_a} \right\} \right] \quad \text{---(63)}$$

which is a generalisation of the binary-state equivalent in equation (50).

4.3. MULTIPLE INSTANCES OF IDENTICAL COMPONENTS

As with the case for binary-state systems explored in chapter 3, multiple occurrences of identical components are incorporated into the likelihood function using *state of knowledge* dependence. The underlying state probabilities of each identical component (i.e. *component type*) are the same, and modify the likelihood function accordingly. The unknown of interest, \mathbf{p} , then becomes the set of *component type state probabilities* where n is the number of *component types*. The total number of components again is denoted n' (and therefore $n \leq n'$). The *component types* are numbered $1, 2, 3, \dots, j, \dots, n$, and the *components* are numbered $1, 2, 3, \dots, b, \dots, n'$. The *component type* of the b^{th} component is denoted j_b . The likelihood function of a multi-state on-demand system with multiple occurrences of the same component is equation (64).

$$L(E | \mathbf{p}) \propto \sum_{\forall \tilde{\mathbf{v}}_a \in \mathbf{v}_E} \left[\prod_{l=1}^{z^n} \left\{ \frac{1}{(v_l)_a!} \left(\prod_{b=1}^{n'} p_{j_b}^{((x_b)_l)} \right)^{(v_l)_a} \right\} \right] \quad \text{---(64)}$$

where $p_{j_b}^{((x_b)_l)}$ is the probability of the b^{th} component being in the $(x_b)_l^{\text{th}}$ state, n' is the number of components in the system and $(x_b)_l$ is the *state variable* of the b^{th} component in the l^{th} *state vector*.

4.4. COMPARISON WITH NON-OVERLAPPING DATA METHODS – MULTI-STATE ON-DEMAND SYSTEMS

The Graves et al. method cannot incorporate information inherent in *overlapping data* sets as it is limited to treating data as *non-overlapping*. *Non-overlapping data* sets contain no such dependence, and therefore there is no information inherently stored in this way. The Graves et al. method is faster to evaluate as it does not involve generation of combinations. Whenever analysis only involves *non-overlapping data*, the Graves et al. method should be employed.

Example 12: *Overlapping data* analysis of a multi-state on-demand system

Consider the basic two component series system examined in Example 5 that was subjected to the same test regime with the same evidence. The structure functions for the system at each sensor location are:

$$\phi_1^S(\tilde{x}) = \phi_1^S(\{x_1, x_2\}) = \max(x_1, x_2) \quad \text{---(65)}$$

$$\phi_2^S(\tilde{x}) = \phi_2^S(\{x_1, x_2\}) = x_1 \quad \text{---(66)}$$

Since there are two components (i.e. $n = 2$) and four possible states (i.e. $z = 4$), the number of possible state vectors is $z^n = 4^2 = 16$. The truncated list of **state vectors** is in Table 7, along with the sensor states they imply and their inherent probabilities.

| State Vector # l | Component States | | State Vector \tilde{x}_l | States detected by sensors | | Probability $\Pr(\tilde{\mathbf{x}} = \tilde{\mathbf{X}}_l \mathbf{p})$ |
|-----------------------|------------------|-----------|-------------------------------|-------------------------------|---------------------------|--|
| | $(x_1)_l$ | $(x_2)_l$ | | $\phi_{s_1}(\tilde{x}_l)$ | $\phi_{s_2}(\tilde{x}_l)$ | |
| 1 | 3 | 3 | {3,3} | 3 | 3 | $p_1^{(3)} p_2^{(3)}$ |
| 2 | 3 | 2 | {3,2} | 3 | 3 | $p_1^{(3)} p_2^{(2)}$ |
| 3 | 3 | 1 | {3,1} | 3 | 3 | $p_1^{(3)} p_2^{(1)}$ |
| 4 | 3 | 0 | {3,0} | 3 | 3 | $p_1^{(3)} p_2^{(0)}$ |
| 5 | 2 | 3 | {2,3} | 3 | 2 | $p_1^{(2)} p_2^{(3)}$ |
| ... | | | | | | |
| 12 | 1 | 0 | {1,0} | 1 | 1 | $p_1^{(1)} p_2^{(0)}$ |
| 13 | 0 | 3 | {0,3} | 3 | 0 | $p_1^{(0)} p_2^{(3)}$ |
| 14 | 0 | 2 | {0,2} | 2 | 0 | $p_1^{(0)} p_2^{(2)}$ |
| 15 | 0 | 1 | {0,1} | 1 | 0 | $p_1^{(0)} p_2^{(1)}$ |
| 16 | 0 | 0 | {0,0} | 0 | 0 | $p_1^{(0)} p_2^{(0)}$ |

Table 7: Truncated list of *state vectors* of system in Figure 1.

Since there are 10 demands (i.e. $r = 10$) there are 3 268 760 possible *state vector combinations*, $\tilde{\mathbf{v}}$, as each demand will invoke one of the 16 possible *state vectors*. A truncated list of combinations is listed in Table 8, along with the implied sensor detection states. There are 600 possible combinations that imply the evidence. The method in which they are compiled is the subject of chapter 5.

The posterior distribution of the component state probabilities was evaluated using Markov-Chain Monte Carlo (MCMC) simulation. The resultant marginal distributions of each state probability are illustrated in Figure 26. They are also compared to the marginal posterior

distributions generated by the Graves et al. (non-overlapping) method in Example 5.

| State Vector Combination No ... a | | 1 | 2 | ... | 602 529 | 602 530 | ... | 3 268 759 | 3 268 760 | |
|--|------------------------------------|----|----|-----|---------|---------|-----|-----------|-----------|----|
| State Vector Combination (no. of l^{th} state vectors) | $(v_1)_a$ | 0 | 0 | ... | 0 | 0 | ... | 9 | 10 | |
| | $(v_2)_a$ | 0 | 0 | ... | 0 | 0 | ... | 1 | 0 | |
| | $(v_3)_a$ | 0 | 0 | ... | 0 | 0 | ... | 0 | 0 | |
| | $(v_4)_a$ | 0 | 0 | ... | 3 | 3 | ... | 0 | 0 | |
| | $(v_5)_a$ | 0 | 0 | ... | 0 | 0 | ... | 0 | 0 | |
| | $(v_6)_a$ | 0 | 0 | ... | 0 | 0 | ... | 0 | 0 | |
| | $(v_7)_a$ | 0 | 0 | ... | 0 | 1 | ... | 0 | 0 | |
| | $(v_8)_a$ | 0 | 0 | ... | 4 | 3 | ... | 0 | 0 | |
| | $(v_9)_a$ | 0 | 0 | ... | 0 | 0 | ... | 0 | 0 | |
| | $(v_{10})_a$ | 0 | 0 | ... | 0 | 0 | ... | 0 | 0 | |
| | $(v_{11})_a$ | 0 | 0 | ... | 0 | 0 | ... | 0 | 0 | |
| | $(v_{12})_a$ | 0 | 0 | ... | 0 | 0 | ... | 0 | 0 | |
| | $(v_{13})_a$ | 0 | 0 | ... | 0 | 0 | ... | 0 | 0 | |
| | $(v_{14})_a$ | 0 | 0 | ... | 0 | 0 | ... | 0 | 0 | |
| | $(v_{15})_a$ | 0 | 1 | ... | 1 | 1 | ... | 0 | 0 | |
| | $(v_{16})_a$ | 10 | 9 | ... | 2 | 2 | ... | 0 | 0 | |
| Implied Evidence | Sensor #1 detected States | 0 | 10 | 9 | ... | 2 | 2 | ... | 0 | 0 |
| | | 1 | 0 | 1 | ... | 1 | 1 | ... | 0 | 0 |
| | | 2 | 0 | 0 | ... | 4 | 4 | ... | 0 | 0 |
| | | 3 | 0 | 0 | ... | 3 | 3 | ... | 10 | 10 |
| | Sensor #2 detected States | 0 | 10 | 10 | ... | 3 | 3 | ... | 0 | 0 |
| | | 1 | 0 | 0 | ... | 0 | 0 | ... | 0 | 0 |
| | | 2 | 0 | 0 | ... | 4 | 4 | ... | 0 | 0 |
| | | 3 | 0 | 0 | ... | 3 | 3 | ... | 10 | 10 |
| 1 if $\tilde{\mathbf{v}}_a \in \mathbf{v}_E$; 0 otherwise | | 0 | 0 | ... | 1 | 1 | ... | 0 | 0 | |

Table 8: Possible state vector combinations of system in Figure 1.

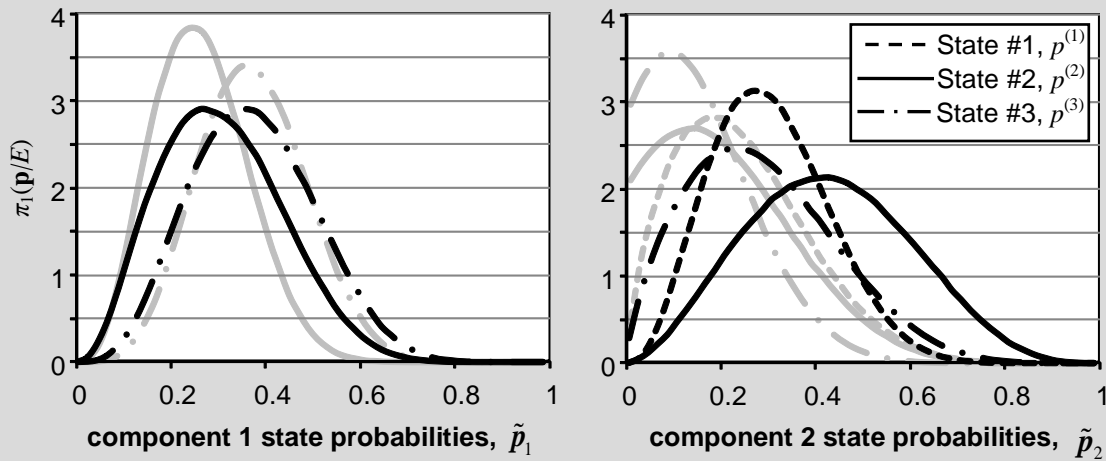


Figure 26: Marginal posterior distributions of each component state probability of the system in Figure 1 (the distributions coloured grey are those derived by the Graves et al. method illustrated in Example 5).

Figure 26 illustrates a marked difference between the posterior distributions produced by the Graves et al and *overlapping data* methods. The reason for this discrepancy stems from the nature of *overlapping data*, and is explored in Figure 27 and Figure 28.

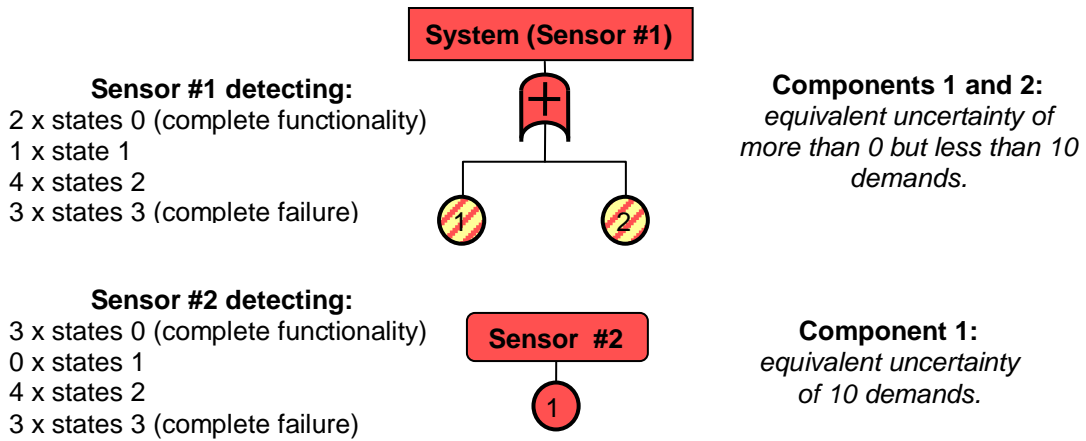


Figure 27: Evidence inferred information when considered as non-overlapping

When treated as *overlapping*, the data sets above cannot achieve the separation as illustrated in Figure 27 but is represented in Figure 28.

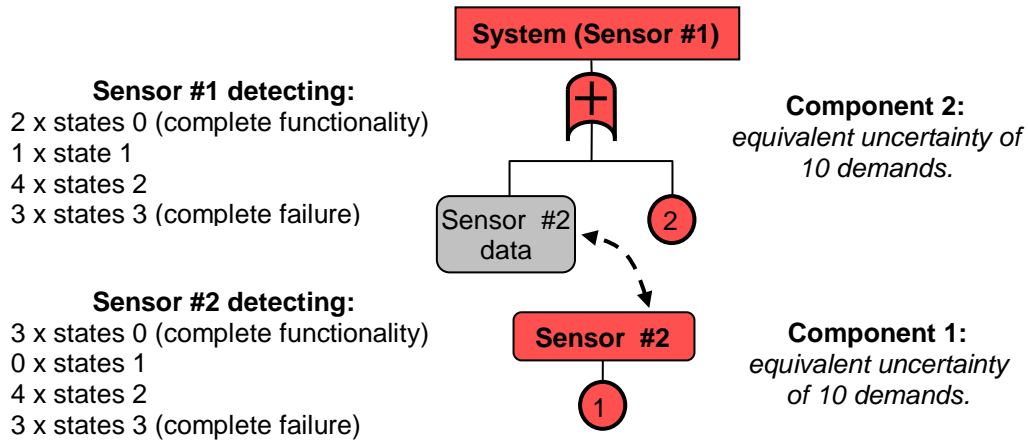


Figure 28: Evidence inferred information when considered as overlapping

Whilst entropy estimates for each marginal posterior distribution have been calculated and shown in Table 9, it is useful to consider the ‘amount of inference’ that each evidence set imparts on each component. For example, a component subjected to 10 demands can be said to have the ‘equivalent uncertainty inferred by 10 demands’. However, a two component system has information ‘shared’ between components, and therefore each component has the ‘equivalent uncertainty inferred by more than 0 but less than 10 demands’.

Even though the characteristics of sensor #1 evidence are a function of both component 1 and 2, the fact that sensor #2 information exists means that the uncertainty implied by sensor #1 evidence is not ‘shared’ between components and is directly imparted onto component 2. A summary of the level of information inferred by the evidence is listed in Table 9.

| | State Prob. | Non-overlapping data | | Overlapping data | | Comparison |
|-------------|-------------|----------------------|----------------------------------|----------------------|----------------------------------|---|
| | | Marg. Entropy (nats) | Equivalent Information (demands) | Marg. Entropy (nats) | Equivalent Information (demands) | |
| Component 1 | $p_1^{(2)}$ | -1.8114 | 10-20 (total) | -1.5029 | 10 | The lower marginal entropies and equivalent uncertainty shows that component 1 characteristics have less uncertainty when data is constrained as non-overlapping |
| | $p_1^{(3)}$ | | | | | |
| Component 2 | $p_2^{(1)}$ | -2.3099 | 0-10 | -2.6852 | 10 | The (generally) higher marginal entropies and equivalent uncertainty shows that component 1 characteristics have more uncertainty when data is constrained as non-overlapping |
| | $p_2^{(2)}$ | | | | | |
| | $p_2^{(3)}$ | | | | | |

Table 9: Comparison of data inference when data is constrained to be non-overlapping (Graves et al. method) versus overlapping (downwards inference) for analysis of system examined in Example 12.

A marked difference in the posterior distributions inferred by the same evidence when considered to be *non-overlapping* versus *overlapping* can be clearly observed. Treating *overlapping data* as *non-overlapping* incorrectly increases the amount of apparent information for the characteristics of some components in some instances, and conversely incorrectly decreases the amount of apparent information for others.

4.5. EVALUATION

The solution of the *overlapping data* likelihood function in equations (53) and (64) cannot easily be carried out by hand even for small systems, and a detailed algorithm (which is the subject of chapter 5) is required for systems of moderate to high complexity. The speed of evaluation is largely dependent on the generation of possible state vectors for the system in question and then identification of all combinations of those state vectors that imply the evidence. Once the state vector combinations are developed, the likelihood function is easily calculated. It must be repeated here that should the data for analysis be *non-overlapping* in nature, the Graves et al. method should be used to save computational time.

4.6. SUMMARY

Whilst fully Bayesian methodologies have been developed for data at various levels within on-demand systems, they generally constrain data to be treated as *non-overlapping*. The only previous methodology to incorporate *overlapping data* is limited to binary-state on-demand systems, with additional limitations discussed in chapter 3. This chapter detailed the generation of the likelihood function for *overlapping data* analysis of multi-state on-demand systems, which is a generalisation of the likelihood function developed in chapter 3. An example was included to highlight the effect of the additional information *overlapping data* contains, how this information can be used to correctly improve our state of knowledge (which is the set of *component type* failure probabilities) and how analysing *overlapping data* as *non-overlapping* sometimes incorporates artificial information. Through state of knowledge dependence, the resultant

overlapping data Bayesian method completely incorporates all information and evidence that can possible be generated or observed by complex, on-demand multi-state systems.

Chapter 5: Downwards Inference Algorithm: Evaluation of *overlapping data* likelihood function of on-demand systems

5.1. INTRODUCTION

The likelihood functions for both binary-state and multi-state on-demand systems developed in chapters 3 and 4 are centred on the generation of a set of combinations of state vectors that infer the observed evidence, and then evaluating the summed probability of observing any one of these possible combinations (noting that they are mutually exclusive). The number of possible combinations that must be considered increases exponentially as the number of components and possible states increases, significantly affecting computational time. An algorithm is outlined in this chapter that allows rapid compilation of this set of combinations of state vectors and hence rapid analysis of the likelihood function for subsequent Bayesian analysis.

5.2. MULTI-STATE ON-DEMAND COMPLEX SYSTEMS: LIKELIHOOD FUNCTION

The likelihood function of observing evidence for a multi-state on-demand system is shown in equation (64).

$$L(E|\mathbf{p}) \propto \sum_{\forall \tilde{\mathbf{v}}_a \in \mathbf{v}_E} \left[\prod_{l=1}^{z^n} \left\{ \frac{1}{(v_l)_a!} \left(\prod_{b=1}^{n'} p_{j_b}^{((x_b)_l)} \right)^{(v_l)_a} \right\} \right] \quad \text{---(64)}$$

where n is the number of component types, n' is the number of components in the system, the unknown of interest, $\mathbf{p} = \{\tilde{\mathbf{p}}_1, \tilde{\mathbf{p}}_2, \tilde{\mathbf{p}}_3, \dots, \tilde{\mathbf{p}}_j, \dots, \tilde{\mathbf{p}}_n\}$ is the set of n lower level component state probability sets for z states, $\tilde{\mathbf{p}}_j = \{p_j^{(1)}, p_j^{(2)}, p_j^{(3)}, \dots, p_j^{(z-1)}\}$ is the state probability vector of the j^{th} component, $p_{j_b}^{((x_b)_l)}$ is the probability of the b^{th} component (which is the j_b^{th} component type) being in the $(x_b)_l^{\text{th}}$ state, r is the number of demands in the data/evidence set E , $\tilde{\mathbf{v}}_a$ is the a^{th} combination of r state vectors (each state vector comprises of n' component states), \mathbf{v}_E is the set of all $\tilde{\mathbf{v}}_a$ that imply the data/evidence set E , $(v_l)_a$ is the number of occurrences of the l^{th} state vector in $\tilde{\mathbf{v}}_a$ and $(x_b)_l$ is the state variable of the b^{th} component in the l^{th} state vector.

The generation of \mathbf{v}_E , the set of combinations of state vectors that imply the evidence, is the most computationally intensive part of developing a likelihood function. The vector $\tilde{\mathbf{v}}$ can be described as a combination of $z^{n'}$ possible state vectors or a permutation of the number of occurrences of each possible state vector, $v_1, v_2, \dots, v_l, \dots, v_{z^{n'}}$. The trivial method of generating \mathbf{v}_E involves considering every possible combination of $z^{n'}$ possible state vectors, which is equivalent to considering every possible permutation of $(v_1, v_2, \dots, v_l, \dots, v_{z^{n'}})$. The number of possible combinations is given in equation (67).

$$\text{Number of possible state vector combinations} = \frac{(z^{n'} + r - 1)!}{r! (z^{n'} - 1)!} \quad \text{---(67)}$$

The number given in equation (67) can become prohibitively large to allow combinations to be individually considered for even moderately complex systems. One of the reasons is that it is exponentially dependent on the number of components, n' . The process of evaluating the likelihood function in equation (64) is illustrated in Figure 29. It introduces the *sensor information vector*, which is simply the vector of states observed by all sensors and can be expressed in terms of structure functions of the component state vector, $\tilde{\mathbf{x}}$.

$$\text{i.e. } \text{Sensor information vector} = \tilde{\mathbf{x}}^S = \{x_1^S, x_2^S, \dots, x_i^S, \dots, x_m^S\} = \tilde{\boldsymbol{\phi}}^S(\tilde{\mathbf{x}}) \quad \text{---(68)}$$

where x_i^S is the state detected by the i^{th} sensor, ϕ_i^S is the structure function of the i^{th} sensor such that $x_i^S = \phi_i^S(\tilde{\mathbf{x}})$ and $\tilde{\boldsymbol{\phi}}^S$ is the vector of the structure functions for all sensors such that $\tilde{\boldsymbol{\phi}}^S = \{\phi_1^S, \phi_2^S, \dots, \phi_i^S, \dots, \phi_m^S\}$.

The number of combinations can be limited by considering *sensor information vectors* as opposed to *component state vectors* because each *sensor information vector* has an associated probability that can be evaluated. If m is the number of sensors, the number of possible combinations of *sensor information vectors* in equation (69) is significantly less than that in equation (67) as there will always be fewer sensors than components (i.e. $m \leq n$).

$$\text{Number of possible sensor state vector combinations} = \frac{(z^m + r - 1)!}{r!(z^m - 1)!} \quad \text{---(69)}$$

To save computational time, the set of *sensor information vector* combinations that imply the evidence can initially be compiled. Each *sensor information vector* can then have individual cut

sets compiled that will then allow the probability of each *sensor information vector* combination to be calculated. The resultant likelihood function is given in equation (70).

$$L(E | \mathbf{p}) \propto \sum_{\forall \tilde{\mathbf{v}}_a^S \in \mathbf{v}_E^S} \prod_{l=1}^{l_{\max}} \left\{ \frac{\left[\Pr(\tilde{\mathbf{x}}_l^S | \mathbf{p}) \right]^{(v_l^S)_a}}{(v_l^S)_a!} \right\} \quad \text{---(70)}$$

where $\Pr(\tilde{\mathbf{x}}_l^S | \mathbf{p}) = \sum_{l'=1}^{z^{n'}} \Pr(\tilde{\mathbf{x}}_{l'} \rightarrow \tilde{\mathbf{x}}_l^S | \mathbf{p})$ which is the sum of the probabilities of each cut set that implies the l^{th} *sensor information vector*, $\tilde{\mathbf{x}}_l^S$, $\tilde{\mathbf{v}}_a^S$ is the a^{th} combination of *sensor information vectors* such that $\tilde{\mathbf{v}}_a^S = \{(v_1^S)_a, (v_2^S)_a, \dots, (v_l^S)_a, \dots, (v_{z^n}^S)_a\}$, $(v_l^S)_a$ is the number of occurrences of the l^{th} *sensor information vector* in the a^{th} combination, l_{\max} is the number of possible *sensor information vectors* and \mathbf{v}_E^S is the set of all $\tilde{\mathbf{v}}_a^S$ that imply the evidence.

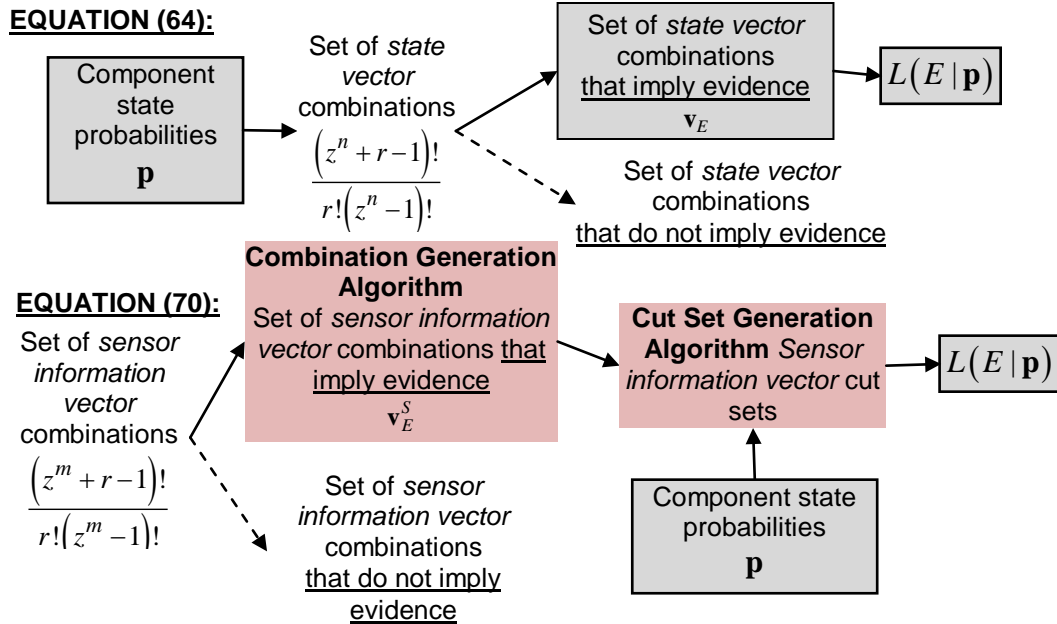


Figure 29: Downwards Inference – Likelihood Function Evaluation Process

These two processes have two separate algorithms developed in the remainder of this chapter.

5.3. MATHEMATICAL REPRESENTATION: EVIDENCE AS A FUNCTION OF SENSOR INFORMATION VECTORS

Formal evaluation of equations (64) and (70) through the processes illustrated in Figure 29 requires evidence to be expressed as a function of *state vectors*. This is best achieved through matrix representation:

$$\mathbf{M} \bullet \tilde{\mathbf{v}}_a^S = \tilde{\mathbf{E}} \quad \text{---(71)}$$

where matrix \mathbf{M} is defined in equation (74) and represents the relationship between $\tilde{\mathbf{v}}_a^S$, the a^{th} combination of sensor information vectors defined in equation (72), and the evidence vector $\tilde{\mathbf{E}}$ defined in equation (73).

$$\tilde{\mathbf{v}}_a^S = \begin{bmatrix} (v_1^S)_a \\ (v_2^S)_a \\ \vdots \\ (v_l^S)_a \\ \vdots \\ (v_{l_{\max}}^S)_a \end{bmatrix} \cdots \quad \text{where } (v_l^S)_a \text{ is the number of times the } l^{\text{th}} \text{ sensor information vector, } \tilde{\mathbf{x}}^S \text{ occurs.} \quad \text{---(72)}$$

The evidence vector, $\tilde{\mathbf{E}}$, contains the sequentially listed number of detections of each of the z states at each sensor. As there are m sensors, $\tilde{\mathbf{E}}$ has a length of $(m \times z)$.

$$\tilde{\mathbf{E}} = \left[\begin{array}{c} \left\{ \begin{array}{c} \text{'0 states' detected by sensor \#1} \\ \vdots \\ \text{'(z-1) states' detected by sensor \#1} \end{array} \right\} \\ \dots \\ \left\{ \begin{array}{c} \text{'0 states' detected sensor \#m} \\ \vdots \\ \text{'(z-1) states' detected sensor \#m} \end{array} \right\} \end{array} \right] = \left[\begin{array}{c} \tilde{\mathbf{k}}_1^S \\ \vdots \\ \tilde{\mathbf{k}}_m^S \end{array} \right] = \left[\begin{array}{c} E_1 \\ E_2 \\ E_3 \\ \vdots \\ E_{[z(i-1)]+x_i^S+1} \\ \vdots \\ E_{(m \times z)} \end{array} \right] = \left[\begin{array}{c} k_1^{S(0)} \\ \vdots \\ k_1^{S(z-1)} \\ \dots \\ k_m^{S(0)} \\ \vdots \\ k_m^{S(z-1)} \end{array} \right] \quad ---(73)$$

The matrix \mathbf{M} is a $(m \times z)$ by l_{\max} matrix (where l_{\max} is the number of possible *sensor information vectors*). Each element, $M_{i,l}$ is either 0 or 1 and relates the effect that the l^{th} *sensor information vector* has on each element of the evidence vector, $\tilde{\mathbf{E}}$. The entire system is therefore summarized by the matrix \mathbf{M} .

$$\mathbf{M} = \left[\begin{array}{cccccc} M_{1,1} & M_{1,2} & \dots & M_{1,l} & \dots & M_{1,l_{\max}} \\ M_{2,1} & M_{2,2} & \dots & M_{2,l} & \dots & M_{2,l_{\max}} \\ \vdots & \vdots & \ddots & \vdots & & \vdots \\ M_{[z(i-1)]+x_i^S+1,1} & M_{[z(i-1)]+x_i^S+1,2} & \dots & M_{[z(i-1)]+x_i^S+1,l} & \dots & M_{[z(i-1)]+x_i^S+1,l_{\max}} \\ \vdots & \vdots & & \vdots & \ddots & \vdots \\ M_{(m \times z),1} & M_{(m \times z),2} & \dots & M_{(m \times z),l} & \dots & M_{(m \times z),l_{\max}} \end{array} \right] \quad ---(74)$$

$$\text{where } \dots M_{[z(i-1)]+x_i^S+1,l} = \begin{cases} 1 \text{ if the } l^{\text{th}} \text{ sensor state vector implies} \\ \quad \text{the } (x_i^S)^{\text{th}} \text{ state at the } i^{\text{th}} \text{ sensor} \\ \dots \\ 0 \text{ otherwise} \end{cases}$$

Example 13: Matrix equation of *sensor information vectors*

Consider the basic two component series multi-state system that is illustrated in Figure 30. There are four allowable states (i.e. $z = 4$) which are 0 (complete functionality), 1 (1st degraded state), 2 (2nd degraded state) and 3 (complete failure).

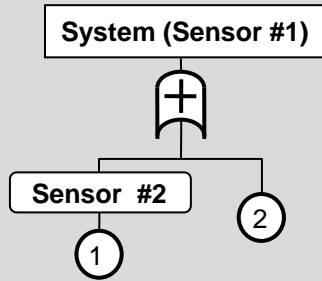


Figure 30: Multi-state two component series system

As the system is a series system, the state detected by sensor #1 must be greater than or equal to the state detected by sensor #2. There are 10 possible *sensor information vectors* and hence $l_{\max} = 10$. The set of 10 possible *sensor information vectors* is listed in equation (75).

$$\begin{Bmatrix} x_1^S = \dots \\ x_2^S = \dots \end{Bmatrix} \Rightarrow \left\{ \begin{Bmatrix} \tilde{x}_1^S \\ 0 \end{Bmatrix}, \begin{Bmatrix} \tilde{x}_2^S \\ 0 \end{Bmatrix}, \begin{Bmatrix} \tilde{x}_3^S \\ 0 \end{Bmatrix}, \begin{Bmatrix} \tilde{x}_4^S \\ 0 \end{Bmatrix}, \begin{Bmatrix} \tilde{x}_5^S \\ 1 \end{Bmatrix}, \begin{Bmatrix} \tilde{x}_6^S \\ 1 \end{Bmatrix}, \begin{Bmatrix} \tilde{x}_7^S \\ 1 \end{Bmatrix}, \begin{Bmatrix} \tilde{x}_8^S \\ 2 \end{Bmatrix}, \begin{Bmatrix} \tilde{x}_9^S \\ 2 \end{Bmatrix}, \begin{Bmatrix} \tilde{x}_{10}^S \\ 3 \end{Bmatrix} \right\} \quad \text{---(75)}$$

Evidence is of the form of the number of observed states at each sensor from r demands.

$$\text{Sensor \#1:} \quad \tilde{\mathbf{k}}_1^S = \{k_1^{S(0)}, k_1^{S(1)}, k_1^{S(2)}, k_1^{S(3)}\} \quad \text{---(76)}$$

$$\text{Sensor \#2:} \quad \tilde{\mathbf{k}}_2^S = \{k_2^{S(0)}, k_2^{S(1)}, k_2^{S(2)}, k_2^{S(3)}\} \quad \text{---(77)}$$

$$\text{Evidence,} \quad \mathbf{E} = \{ \tilde{\mathbf{k}}_1^S, \tilde{\mathbf{k}}_2^S, r \} \quad \text{---(78)}$$

The performance of the system is described by a combination of r *sensor information vectors* from the set in equation (75). The number of occurrences of the *sensor information vector* $\tilde{\mathbf{x}}_l^S$ is v_l^S . The number of times the sensor #1 detects a ‘0’ state for the a^{th} combination of *sensor information vectors*, $\tilde{\mathbf{v}}_a^S$, is the sum of all *sensor information vectors* in the combination where $x_1^S = 0$ (which is only the first *sensor information vector*, $\tilde{\mathbf{x}}_1^S$).

$$\text{i.e.} \quad k_1^0 = v_1^S \quad \text{---(79)}$$

The number of times the sensor #1 detects a ‘1’ state for the a^{th} combination of *sensor information vectors*, $\tilde{\mathbf{v}}_a^S$, is the sum of all *sensor information vectors* in the combination where $x_1^S = 1$ (which is only the 2nd and 5th *sensor information vectors*, $\tilde{\mathbf{x}}_2^S$ and $\tilde{\mathbf{x}}_5^S$).

$$\text{i.e.} \quad k_1^1 = v_2^S + v_5^S \quad \text{---(80)}$$

This process is repeated and sets up a total of $(z \times m)$ simultaneous equations that can be represented in matrix notation as demonstrated in equation (71).

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \bullet \begin{bmatrix} v_1^S \\ v_2^S \\ v_3^S \\ v_4^S \\ v_5^S \\ v_6^S \\ v_7^S \\ v_8^S \\ v_9^S \\ v_{10}^S \end{bmatrix} = \begin{bmatrix} k_1^0 \\ k_1^1 \\ k_1^2 \\ k_1^3 \\ k_2^0 \\ k_2^1 \\ k_2^2 \\ k_2^3 \end{bmatrix} \quad \text{---(81)}$$

The matrix equation (81) of the form $\mathbf{M} \bullet \tilde{\mathbf{v}}_a^S = \tilde{\mathbf{E}}$ summarizes the relationship between a combination of r *sensor information vectors* and the evidence it implies.

5.4. DOWNWARDS INFERENCE – COMBINATION GENERATION ALGORITHM

To evaluate equation (70), the set of combinations of *sensor information vectors* that imply the evidence need to be compiled. These combinations are of the form:

$$\tilde{\mathbf{v}}^S = \{v_1^S, v_2^S, \dots, v_l^S, \dots, v_{l_{\max}}^S\} \quad \text{---(82)}$$

where v_l^S is the number of occurrences of the l^{th} *sensor information vector*, $\tilde{\mathbf{x}}_l^S$ in the *sensor information vector* combination, $\tilde{\mathbf{v}}^S$.

The *combination generation algorithm* proposed below allows for rapid compilation of such sets, and involves the following steps.

5.4.1. Step 1: Generation of Constraints.

The set of requirements that each combination must satisfy is subsequently referred to as *constraints* (such as ' \tilde{v}^S must imply the evidence'). Another *constraint* is that the sum of all v_l values must equal the number of demands, r . There are two types of constraints:

- a. Partial (optional, but will greatly speed computation). *Partial constraints* are those that apply to *combination sub-sets* $\{v_1^S, v_2^S, \dots, v_l^S\} \dots l < l_{max}$. If any *combination sub-set* violates a constraint, then all *complete combinations* that contain that sub-set can be discarded.
- b. Total. *Total constraints* are those that apply to *complete combinations* $\{v_1^S, v_2^S, \dots, v_l^S, \dots, v_{l_{max}}^S\}$ (i.e. the completely defined combination).

5.4.2. Step 2: Identification of common v_l^S values (optional).

The *constraints* may limit certain instances of v_l^S to specific values. By identifying these instances where v_l^S has a certain common value throughout all possible combinations, the number of subsequent possible combinations is significantly reduced, thereby reducing

computational requirements. One of two possible conditions must be met for common v_l^S values to exist:

- a. Null v_l^S values. If the evidence set involves any values for which $k_i^{S(x_i^S)} = 0$, then all non-zero elements of the matrix \mathbf{M} on the $([z(i-1)] + x_i^S + 1)^{\text{th}}$ row imply that the number of occurrences of the relevant *sensor information vectors*, v_l^S , must also be zero (where l is the column number of all non-zero elements).

$$\text{i.e. if } k_i^{S(x)} = 0 \rightarrow v_l^S = 0 \text{ for all } l \text{ such that } M_{([z(i-1)] + x_i^S + 1), l} = 1. \quad \text{---(83)}$$

- b. Trivial v_l^S values. If the $([z(i-1)] + x_i^S + 1)^{\text{th}}$ row of the matrix \mathbf{M} only has one non-zero value (at the element $M_{([z(i-1)] + x_i^S + 1), l}$), then the number of l^{th} *sensor information vectors* that occur in the combination of *sensor information vectors* is $k_i^{(x_i^S)}$.

$$\text{i.e. if } M_{([z(i-1)] + x_i^S + 1), l} = 1 \text{ and } \forall M_{([z(i-1)] + x_i^S + 1), \neq l} = 0 \rightarrow v_l^S = k_i^{(x_i^S)} \quad \text{---(84)}$$

For each common v_l^S value, the matrix \mathbf{M} and vectors $\tilde{\mathbf{E}}$ and $\tilde{\mathbf{v}}_a^S$ are modified in the following ways:

$$\tilde{\mathbf{E}}' = \tilde{\mathbf{E}} - v_l^S \bullet \begin{bmatrix} M_{1,l} \\ M_{2,l} \\ \vdots \\ M_{[z(i-1)]+x_i^S+1,l} \\ \vdots \\ M_{(m \times z),l} \end{bmatrix} \text{ and } \tilde{\mathbf{v}}_a^{S'} = \begin{bmatrix} (v_1^S)_a \\ (v_2^S)_a \\ \vdots \\ (v_{l-1}^S)_a \\ (v_{l+1}^S)_a \\ \vdots \\ (v_1^S)_a \end{bmatrix} \dots \text{removing the } l^{\text{th}} \text{ element} \text{---(85)(86)}$$

$$\mathbf{M}' = \begin{bmatrix} M_{1,1} & \dots & M_{1,l-1} & M_{1,l+1} & \dots & M_{1,l_{\max}} \\ M_{2,1} & \dots & M_{2,l-1} & M_{2,l+1} & \dots & M_{2,l_{\max}} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ M_{[z(i-1)]+x_i^S+1,1} & \dots & M_{[z(i-1)]+x_i^S+1,l-1} & M_{[z(i-1)]+x_i^S+1,l+1} & \dots & M_{[z(i-1)]+x_i^S+1,l_{\max}} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ M_{(m \times z),1} & \dots & M_{(m \times z),l-1} & M_{(m \times z),l+1} & \dots & M_{(m \times z),l_{\max}} \end{bmatrix} \text{---(87)}$$

removing the l^{th} column where v_l^S is the identified common value.

$$\text{and } r' = r - v_l^S \text{---(88)}$$

Should the matrix \mathbf{M}' have rows where all elements are zero, then that row (along with the corresponding row in the evidence vector $\tilde{\mathbf{E}}'$) is removed. If the row in $\tilde{\mathbf{E}}$ that is removed in this process is a non-zero element, then there is an error in the way the matrix \mathbf{M} models the system.

$$\text{i.e. if } M_{i,l} = 0, \forall l \Rightarrow \dots \mathbf{M}' = \begin{bmatrix} M_{1,1} & M_{1,2} & \cdots & M_{1,l} & \cdots & M_{1,l_{\max}} \\ M_{2,1} & M_{2,2} & \cdots & M_{2,l} & \cdots & M_{2,l_{\max}} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ M_{i-1,1} & M_{i-1,2} & \cdots & M_{i-1,l} & \cdots & M_{i-1,l_{\max}} \\ M_{i+1,1} & M_{i+1,2} & \cdots & M_{i+1,l} & \cdots & M_{i+1,l_{\max}} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ M_{(m \times z),1} & M_{(m \times z),2} & \cdots & M_{(m \times z),l} & \cdots & M_{(m \times z),l_{\max}} \end{bmatrix} \quad \text{---(89)}$$

$$\tilde{\mathbf{E}}' = \begin{bmatrix} E_1 \\ E_2 \\ \vdots \\ E_{i-1} \\ E_{i+1} \\ \vdots \\ E_{m \times z} \end{bmatrix} \quad \text{---(90)}$$

After each cycle, \mathbf{M} becomes \mathbf{M}' , $\tilde{\mathbf{E}}$ becomes $\tilde{\mathbf{E}}'$, $\tilde{\mathbf{v}}_a^S$ becomes $\tilde{\mathbf{v}}_a^{S'}$, and r becomes r' . The process is repeated iteratively until the conditions in (83) and (84) can no longer be satisfied (i.e. no more common v_l^S values exist).

5.4.3. Step 3: Setting minimum and maximum v_l^S values.

One of two possible approaches must be used to establish minimum and maximum v_l^S values:

- a. Trivial. All remaining v_l^S values are assigned values of 0 and r as minimum and maximum values respectively.

$$\{\min_1, \min_2, \dots, \min_l, \dots, \min_{l_{\max}}\} = \{0, 0, \dots, 0, \dots, 0\} \quad \text{---(91)}$$

$$\{\max_1, \max_2, \dots, \max_l, \dots, \max_{l_{\max}}\} = \{r, r, \dots, r, \dots, r\} \quad \text{---(92)}$$

- b. Theoretical (optional, but will greatly speed computation). For a given *partial combination sub-set* $\{v_1^S, v_2^S, \dots, v_l^S\} \dots l < l_{\max}$, v_{l+1}^S has minimum and maximum values derived from *constraints*. The number of subsequent possible combinations is reduced, thereby decreasing computational time. The estimates can be improved iteratively as follows:

| Minimizing \min_l such that: | Maximizing \max_l such that: |
|---|---|
| $\mathbf{M} \bullet \begin{bmatrix} \max_1 \\ \vdots \\ \max_{l-1} \\ \min_l \\ \max_{l+1} \\ \vdots \\ \max_{l_{\max}} \end{bmatrix} \geq \tilde{\mathbf{E}} \quad \text{---(93)}$ | $\mathbf{M} \bullet \begin{bmatrix} \min_1 \\ \vdots \\ \min_{l-1} \\ \max_l \\ \min_{l+1} \\ \vdots \\ \min_{l_{\max}} \end{bmatrix} \leq \tilde{\mathbf{E}} \quad \text{---(94)}$ |

Note: The less than or equals to (\leq) and greater than or equals to (\geq) signs in inequalities (93) and (94) are used to indicate the relationship between all equivalent elements on both the left and right hand sides (i.e. all elements of the resultant vector on the left hand side of (93) ' \geq ' must be greater than or equal to all elements of the vector $\tilde{\mathbf{E}}$ for the inequality hold).

The procedures in (93) and (94) can be repeated iteratively until the two sets in (91) and (92) are developed.

5.4.4. Step 4: Generation of combinations - ‘Sideways consideration / Upwards generation’.

The final process is using the remaining undefined values of v_l^S and their associated minimum and maximum values to generate a set of all combinations that meet the *constraints*. This is done sequentially in sub-sets $\{v_1^S, v_2^S, \dots, v_l^S, \dots, v_{l_{\max}}^S\}$ for $l = 1, 2, \dots, l_{\max}$ (i.e. *sideways consideration*) for each possible remaining combination (i.e. *upwards generation*). Depending on the nature of the set or sub-set under consideration, accepting and rejecting occurs as follows:

- a. Combination Sub-sets $\{v_1^S, v_2^S, \dots, v_l^S\} \dots l < l_{\max}$. *Combination sub-sets* are tested against *partial criteria*. *Partial combination sub-sets* that meet *partial criteria* indicate potential for the remaining undefined values of $\{v_{l+1}^S, v_{l+2}^S, \dots, v_{l_{\max}}^S\}$ to imply the evidence. If any one *partial criterion* is not met, then all combinations that contain that particular sub-set are invalid. This significantly reduces the number of combinations that subsequently need to be considered, reducing computational time.
- b. Total combinations $\{v_1^S, v_2^S, \dots, v_{l_{\max}}^S\}$: *Total combinations* are tested against total criteria. *Total combinations* that meet *total criteria* imply the observed evidence. If they do, then that particular combination is added to the set of combinations that imply the evidence.

The process is represented below in Figure 31.

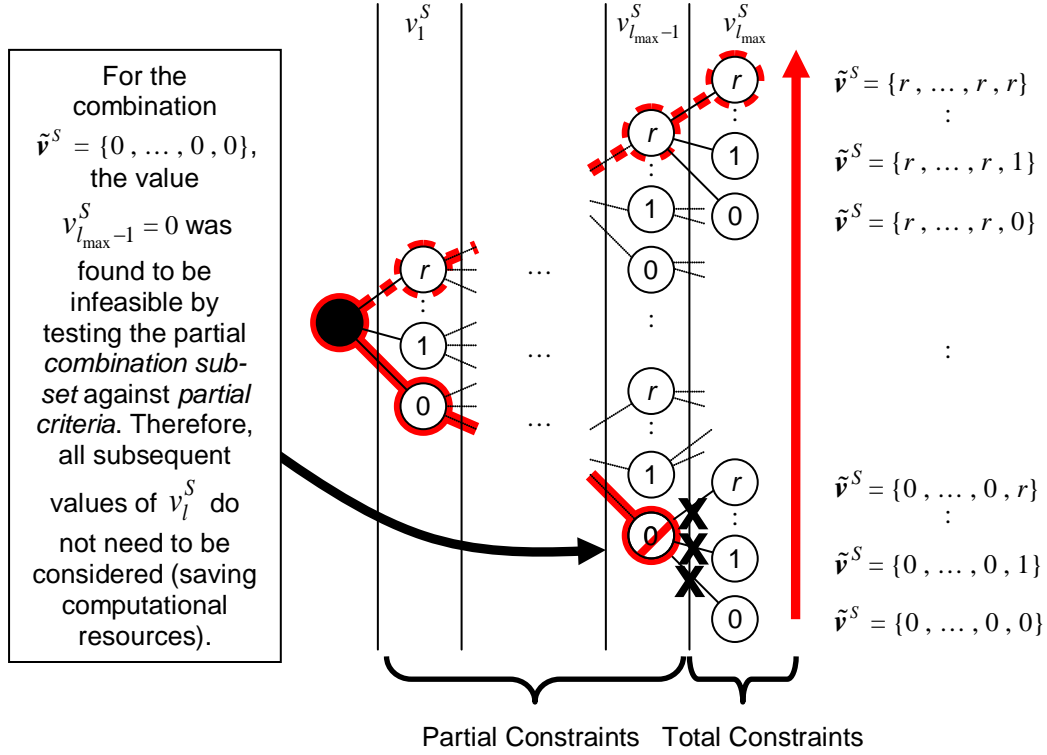


Figure 31: Representation of ‘sideways consideration / upwards generation’ combination algorithm

By identifying infeasible values of v_l^S , large fractions of combinations which have that particular combination sub-set can be excluded from consideration. The earlier this occurs (i.e. the closer l is to 1), the larger the numbers of combinations that can be excluded. This significantly decreases computational time. The process is illustrated in the flow-chart in Figure 32.

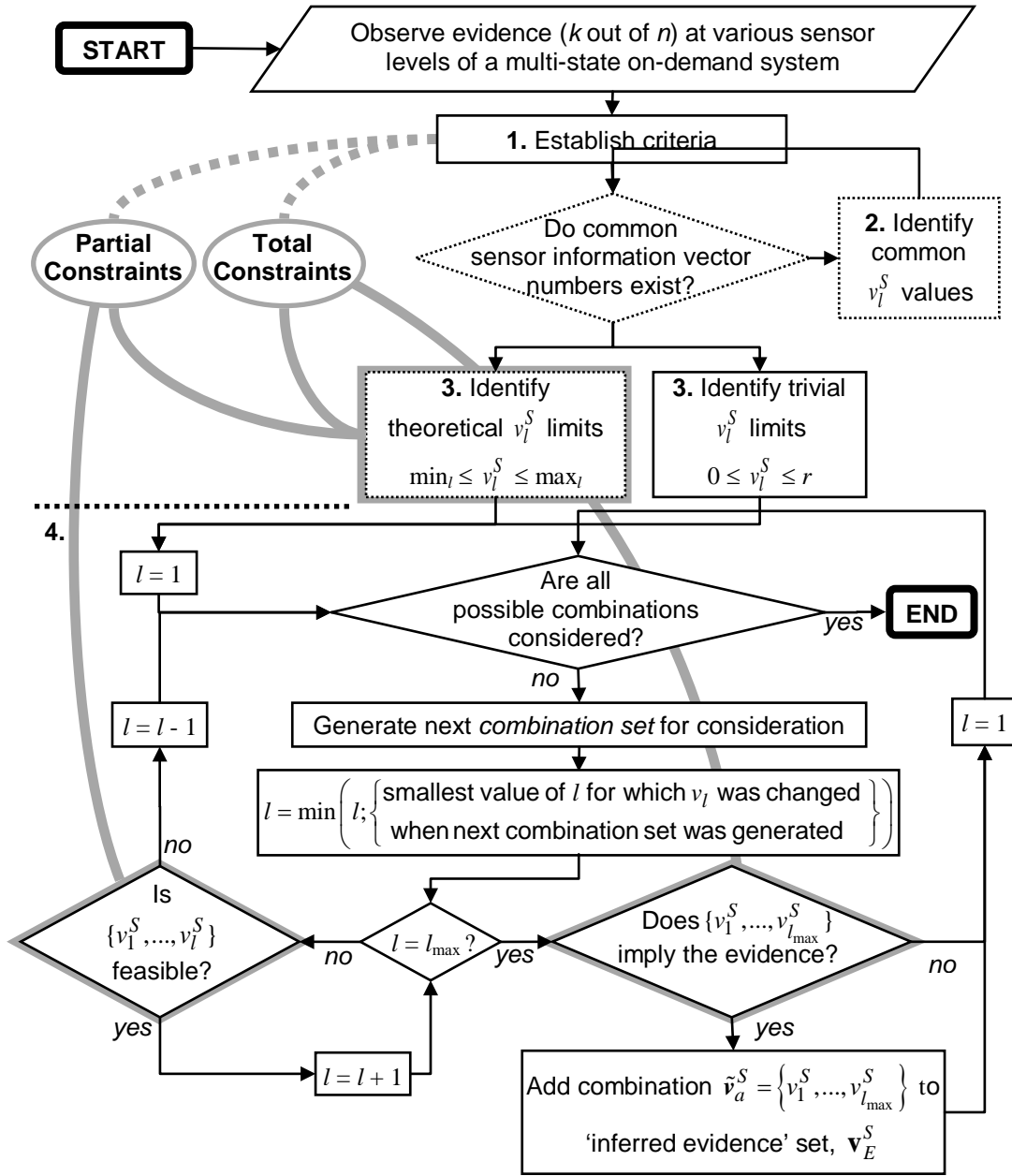


Figure 32: Downwards Inference Combination Generation Algorithm Flow-Chart

(‘sideways consideration / upwards generation’)

Example 14: Solving matrix equation of *sensor information vectors*

Consider the same two component series system in Example 13. The system was subjected to 10 demands (i.e. $r = 10$) and the following evidence was obtained:

$$\text{Sensor \#1:} \quad \tilde{\mathbf{k}}_1^S = \{k_1^{S(0)}, k_1^{S(1)}, k_1^{S(2)}, k_1^{S(3)}\} = \{0, 0, 5, 5\} \quad \text{---(95)}$$

$$\text{Sensor \#2:} \quad \tilde{\mathbf{k}}_2^S = \{k_2^{S(0)}, k_2^{S(1)}, k_2^{S(2)}, k_2^{S(3)}\} = \{0, 3, 4, 3\} \quad \text{---(96)}$$

$$\text{Evidence,} \quad E = \{ \tilde{\mathbf{k}}_1^S, \tilde{\mathbf{k}}_2^S, r \} = \{ \tilde{\mathbf{k}}_1^S, \tilde{\mathbf{k}}_2^S, 10 \} \quad \text{---(97)}$$

Substitution into equation (81) yields:

$$\mathbf{M} \bullet \tilde{\mathbf{v}}_a^S = \tilde{\mathbf{E}} \Rightarrow \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \bullet \begin{bmatrix} v_1^S \\ v_2^S \\ v_3^S \\ v_4^S \\ v_5^S \\ v_6^S \\ v_7^S \\ v_8^S \\ v_9^S \\ v_{10}^S \end{bmatrix} = \begin{bmatrix} k_1^{S(0)} \\ k_1^{S(1)} \\ k_1^{S(2)} \\ k_1^{S(3)} \\ k_2^{S(0)} \\ k_2^{S(1)} \\ k_2^{S(2)} \\ k_2^{S(3)} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 5 \\ 5 \\ 0 \\ 3 \\ 4 \\ 3 \end{bmatrix} \quad \text{---(98)}$$

Since there were 10 demands, it can also be written:

$$\sum_{l=1}^{10} v_l^S = r = 10 \quad \text{---(99)}$$

Equations (98) and (99) together form the ***total criteria*** for the evidence based on the system logic. Both equations (98) and (99) must be met for the *combination of sensor information vectors*, $\{v_1^S, v_2^S, \dots, v_l^S, \dots, v_{10}^S\}$ to be valid.

Partial criteria deal with subsets of the *combination of sensor information vectors*, $\{v_1^S, v_2^S, \dots, v_l^S\}$ where $l < 10$. The fact that v_l^S is a non negative integer can be exploited to establish ***partial criteria***. Let $\{\min_1, \min_2, \dots, \min_l, \dots, \min_{10}\}$ and $\{\max_1, \max_2, \dots, \max_l, \dots, \max_{10}\}$ bet the set of minimum and maximum values respectively for all v_l^S . This allows the following equations to be written to express the ***partial criteria***:

$$\mathbf{M} \bullet \begin{bmatrix} v_1^S \\ \vdots \\ v_l^S \\ \min_{l+1} \\ \vdots \\ \min_{10} \end{bmatrix} \leq \tilde{\mathbf{E}} \quad \text{and} \quad \mathbf{M} \bullet \begin{bmatrix} v_1^S \\ \vdots \\ v_l^S \\ \max_{l+1} \\ \vdots \\ \max_{10} \end{bmatrix} \geq \tilde{\mathbf{E}} \quad \text{where} \quad \tilde{\mathbf{E}} = \begin{bmatrix} k_1^{S(0)} \\ k_1^{S(1)} \\ k_1^{S(2)} \\ k_1^{S(3)} \\ k_2^{S(0)} \\ k_2^{S(1)} \\ k_2^{S(2)} \\ k_2^{S(3)} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 5 \\ 5 \\ 0 \\ 3 \\ 4 \\ 3 \end{bmatrix} \quad \text{---(100)(101)}$$

$$\sum_{l'=1}^l v_{l'}^S + \sum_{l'=l+1}^{10} \min_{l'} \leq r = 10 \quad \text{and} \quad \sum_{l'=1}^l v_{l'}^S + \sum_{l'=l+1}^{10} \max_{l'} \geq r = 10 \quad \text{---(102)(103)}$$

The next step is to identify ***common sensor information vector numbers***, v_l^S . From equation (83), it can be observed that $v_1^S, v_2^S, v_3^S, v_4^S$ and v_5^S are equal to 0 as corresponding non-zero

elements exist on rows 1, 2 and 5 which have zero observations in the evidence vector, \tilde{E} . From equation (84), it can be observed that $v_{10}^S = k_2^{S(3)} = 3$ since the only non-zero element of the 8th row of matrix \mathbf{M} corresponds to v_{10}^S , and the 8th row of the evidence vector, \tilde{E} , is $k_2^{S(3)}$.

$$\mathbf{M} = \begin{bmatrix} \boxed{1} & \boxed{0} & \boxed{0} & \boxed{0} & \boxed{0} & \boxed{0} & \boxed{0} & \boxed{0} & \boxed{0} & \boxed{0} \\ 0 & \boxed{1} & \boxed{0} & \boxed{0} & \boxed{1} & \boxed{0} & \boxed{0} & \boxed{0} & \boxed{0} & \boxed{0} \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ \boxed{1} & \boxed{1} & \boxed{1} & \boxed{1} & \boxed{0} & \boxed{0} & \boxed{0} & \boxed{0} & \boxed{0} & \boxed{0} \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ \cancel{0} & \cancel{0} & \cancel{0} & \cancel{0} & \cancel{0} & \cancel{0} & \cancel{0} & \cancel{0} & \cancel{0} & \boxed{1} \end{bmatrix} \begin{matrix} \leftarrow \\ \leftarrow \\ \leftarrow \\ \leftarrow \\ \leftarrow \\ \leftarrow \\ \leftarrow \\ \leftarrow \end{matrix} \begin{bmatrix} \boxed{0} \\ \boxed{0} \\ 5 \\ 5 \\ \boxed{0} \\ 3 \\ 4 \\ \boxed{3} \end{bmatrix} \quad \dots \quad \tilde{E} =$$

With these common values, equations (85) to (89) reduce the matrix equation to:

$$\mathbf{M}' \bullet \tilde{\mathbf{r}}_a^{S'} = \tilde{\mathbf{E}}' \rightarrow \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \bullet \begin{bmatrix} v_6^S \\ v_7^S \\ v_8^S \\ v_9^S \end{bmatrix} = \begin{bmatrix} 5 \\ 2 \\ 3 \\ 4 \end{bmatrix} \quad \text{---(104)}$$

where $v_1^S = v_2^S = v_3^S = v_4^S = v_5^S = 0$, $v_{10}^S = 3$ and $r' = 7$.

The sets $\{\min_6, \min_7, \min_8, \min_9\}$ and $\{\max_6, \max_7, \max_8, \max_9\}$ are **trivially** set at:

$$\{\min_6, \min_7, \min_8, \min_9\} = \{0, 0, 0, 0\} \quad \text{---(105)}$$

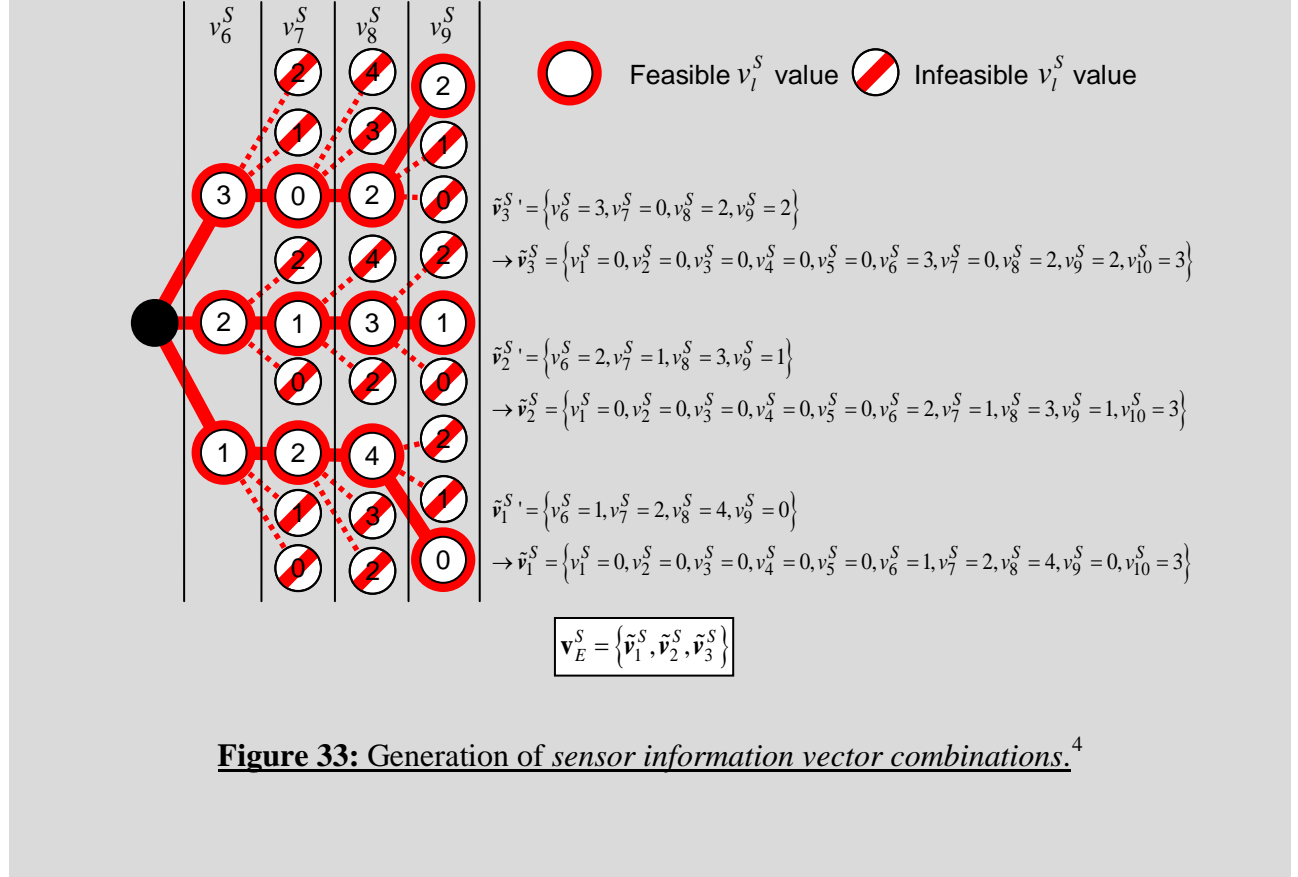
$$\{\max_6, \max_7, \max_8, \max_9\} = \{r', r', r', r'\} \quad \text{---(106)}$$

The minimum and maximum sets can be improved based on the matrix **M** and evidence set *E* through iteratively undertaking (93) and (94) until the sets in (105) and (106) reach steady state. Accordingly, the minima and maxima are as follows:

$$\{\min_6, \min_7, \min_8, \min_9\} = \{1, 0, 2, 0\} \quad \text{---(107)}$$

$$\{\max_6, \max_7, \max_8, \max_9\} = \{3, 2, 4, 2\} \quad \text{---(108)}$$

The **generation of combinations** is then carried out using the **partial criteria** represented in equations (100), (101), (102) and (103), and **total criteria** represented in equations (99) and (104). The process is illustrated in Figure 33. It can be seen that there are three *sensor information vector combinations* that imply the evidence. The method illustrated in Figure 31 and Figure 32 has reduced the problem to that of one considering 9 different combinations of 4 *sensor information vectors* as opposed to the completely trivial method of individual consideration of a possible 3 724 680 960 combinations of 10 *sensor information vectors*, allowing simple numerical compilation in negligible computing time.



5.5. DOWNWARDS INFERENCE - CUT SET GENERATION ALGORITHM

To evaluate the likelihood function, the probabilities of the *sensor information vector combinations* that have been generated by the *Combination Generation Algorithm* outlined above need to be calculated. The probability of each component state probability cut set (where each cut set is in effect a specific instance of a *component state vector*) is easily evaluated. Each *sensor state combination* will have a corresponding set of *component cut sets*, $\tilde{\mathbf{x}} = \{x_1, x_2, \dots, x_b, \dots, x_n\}$, and can be substituted into equation (70) to yield the *sensor information vector probability*.

⁴ The author has observed that in all the instances he has run this algorithm, the corresponding illustration to figure 5 always generates a symmetric path through feasible combination elements.

The *Cut Set Generation Algorithm* process is similar to the *Combination Generation Algorithm* in that it rapidly eliminates large numbers of *component cut sets* for consideration by rapidly identifying and discarding those that are irrelevant. There are a total of $z^{n'}$ possible sets (or *component state vectors*) with only a small fraction of them relevant cut sets. From the *Combination Generation Algorithm* a set of ‘relevant’ *sensor information vectors* will be identified as all *sensor information vectors* from which all *combinations of sensor information vectors* consist of.

$$\text{i.e. } \mathbf{x}^S = \left\{ \forall \tilde{\mathbf{x}}_l^S \right\} \text{ for } \forall l \text{ where } v_l^S \neq 0 \text{ for each combination of sensor state vectors} \quad \text{---(109)}$$

5.5.1. Step 1: Generation of Constraints.

The set of requirements that each cut set must satisfy is subsequently referred to as *constraints*.

The two types of constraints introduced earlier still apply:

- a. Partial (optional, but will greatly speed computation). *Partial constraints* are those that apply to *cut sub-sets* $\{x_1, x_2, \dots, x_b\}$, $b < n'$. In this instance, a *partial constraint* implies that a particular *cut sub-set* belongs to at least one *complete cut set* that implies one of the *sensor information vectors* considered in the *Combination Generation Algorithm*.
- b. Total. *Total constraints* are those that apply to *complete cut sets* $\tilde{\mathbf{x}} = \{x_1, x_2, \dots, x_b, \dots, x_{n'}\}$ (i.e. the completely defined *cut set*). In this instance, a *total*

constraint implies that a particular *complete cut set* implies one of the *sensor information vectors* considered in the *Combination Generation Algorithm*.

5.5.2. Step 2: Setting minimum and maximum component state variables, x_j .

One of two possible approaches must be used to establish minimum and maximum x_j values:

- a. Trivial. All component state variables. x_b , are assigned values of 0 and $(z-1)$ as minimum and maximum values respectively.

$$\{\min_1, \min_2, \dots, \min_b, \dots, \min_n\} = \{0, 0, \dots, 0, \dots, 0\} \quad \text{---(110)}$$

$$\{\max_1, \max_2, \dots, \max_b, \dots, \max_n\} = \{(z-1), (z-1), \dots, (z-1), \dots, (z-1)\} \quad \text{---(111)}$$

- b. Theoretical (optional, but will greatly speed computation). All component state variables. x_b , have minimum and maximum values derived from *constraints*. The number of subsequent possible *cut sets* is significantly reduced, thereby reducing computational timeframes. The estimates can be improved iteratively as follows:

Minimizing \min_b such that:

$$\tilde{\mathbf{x}}^S = \begin{Bmatrix} x_1^S \\ x_2^S \\ \dots \\ x_i^S \\ \dots \\ x_m^S \end{Bmatrix} = \begin{Bmatrix} \phi_1^S(\{\max_1, \dots, \max_{b-1}, \min_b, \max_{b+1}, \dots, \max_n\}) \\ \phi_2^S(\{\max_1, \dots, \max_{b-1}, \min_b, \max_{b+1}, \dots, \max_n\}) \\ \dots \\ \phi_i^S(\{\max_1, \dots, \max_{b-1}, \min_b, \max_{b+1}, \dots, \max_n\}) \\ \dots \\ \phi_m^S(\{\max_1, \dots, \max_{b-1}, \min_b, \max_{b+1}, \dots, \max_n\}) \end{Bmatrix} \geq \tilde{\mathbf{x}}^S \quad \text{---(112)}$$

and maximizing \max_b such that:

$$\tilde{\mathbf{x}}^S = \begin{Bmatrix} x_1^S \\ x_2^S \\ \dots \\ x_i^S \\ \dots \\ x_m^S \end{Bmatrix} = \begin{Bmatrix} \phi_1^S(\{\min_1, \dots, \min_{b-1}, \max_b, \min_{b+1}, \dots, \min_n\}) \\ \phi_2^S(\{\min_1, \dots, \min_{b-1}, \max_b, \min_{b+1}, \dots, \min_n\}) \\ \dots \\ \phi_i^S(\{\min_1, \dots, \min_{b-1}, \max_b, \min_{b+1}, \dots, \min_n\}) \\ \dots \\ \phi_m^S(\{\min_1, \dots, \min_{b-1}, \max_b, \min_{b+1}, \dots, \min_n\}) \end{Bmatrix} \leq \tilde{\mathbf{x}}^S \quad \text{---(113)}$$

$$\text{where } \tilde{\mathbf{x}}^S = \begin{Bmatrix} x_1^S \\ x_2^S \\ \dots \\ x_i^S \\ \dots \\ x_m^S \end{Bmatrix}, \quad \tilde{\mathbf{x}}^S \in \mathbf{x}^S.$$

Equations (112) and (113) are repeated until the two sets in (110) and (111) are developed.

5.5.3. Step 3: Generation of cut sets - 'Sideways consideration / Upwards generation'.

As previously discussed, the final process is using the remaining undefined values of x_j and their associated minimum and maximum values to generate a set of all cut sets that meet the *constraints*. This is done sequentially in sub-sets $\{x_1, x_2, \dots, x_b\}$, for $b = 1, 2, \dots, n'$ (i.e. *sideways consideration*) for each possible remaining combination (i.e. *upwards generation*).

Acceptance and rejection occurs as follows:

- a. Cut Sub-sets $\{x_1, x_2, \dots, x_b\}, b < n'$. *Cut sub-sets* are tested against *partial criteria*. *Cut sub-sets* that meet *partial criteria* indicate potential for the remaining undefined values of $\{x_{b+1}, x_{b+2}, \dots, x_{n'}\}$ to imply the evidence. If any one *partial criterion* is not met, then all combinations that contain that particular sub-set are invalid. This significantly reduces the number of possible cut sets that subsequently need to be considered, reducing computational time.
- b. Complete Cut Sets $\{x_1, x_2, \dots, x_b, \dots, x_{n'}\}$. *Complete cut sets* are tested against total criteria. *Complete cut sets* that meet *total criteria* imply the observed evidence and are added to the set of cut sets that imply the evidence.

The process is illustrated in Figure 34.

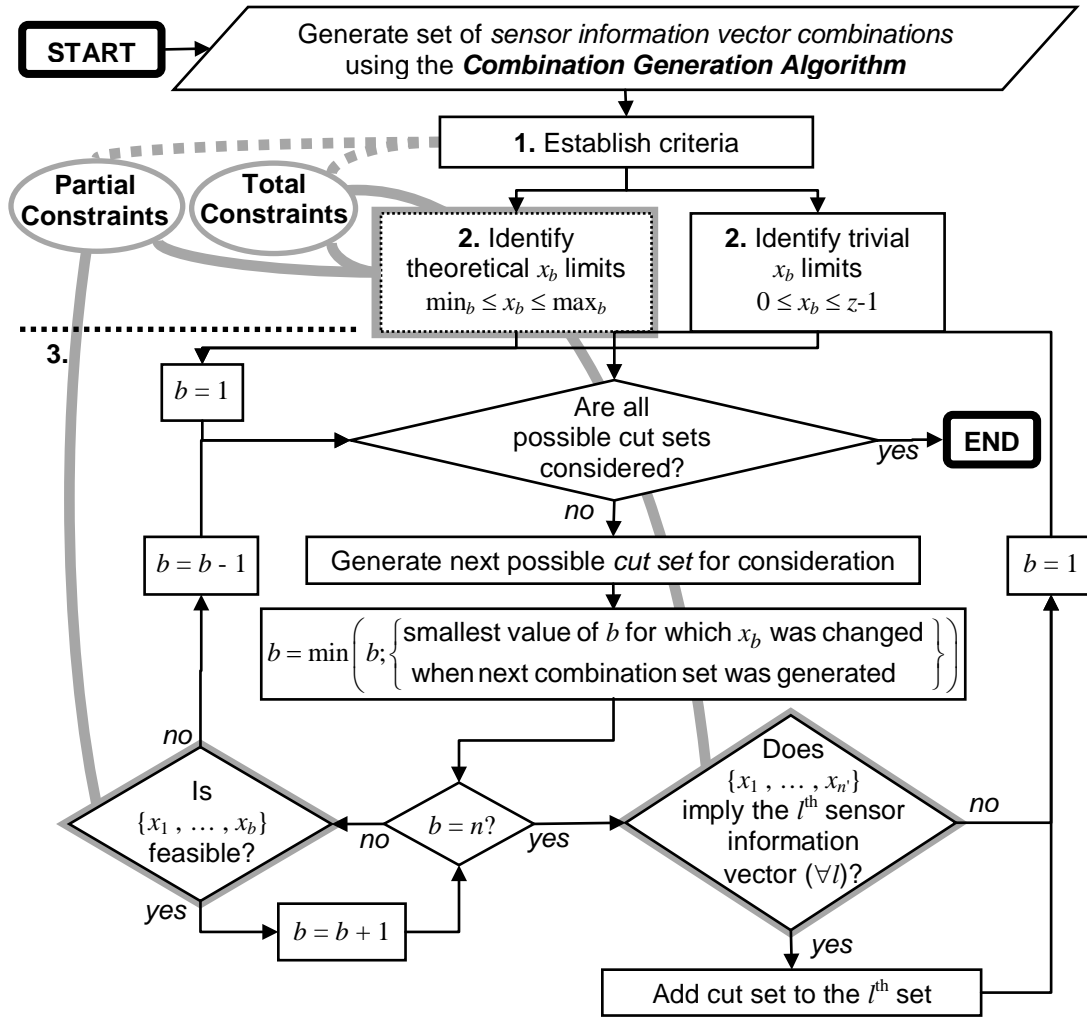


Figure 34: Overlapping Cut Set Generation Algorithm

(‘sideways consideration / upwards generation’)

Example 15: Cut-set (state vector) generation for *sensor information vectors*

Consider the two component series system in Example 14. The *Combination Generation Algorithm* generated three possible combinations of sensor information vectors that implied the evidence, and are listed in equations (114), (115), and (116).

$$\tilde{\mathbf{v}}_1^S = \{v_6^S = 1, v_7^S = 2, v_8^S = 4, v_{10}^S = 3\} \quad \text{---(114)}$$

$$\tilde{\mathbf{v}}_2^S = \{v_6^S = 2, v_7^S = 1, v_8^S = 3, v_9^S = 1, v_{10}^S = 3\} \quad \text{---(115)}$$

$$\tilde{\mathbf{v}}_3^S = \{v_6^S = 3, v_8^S = 2, v_9^S = 2, v_{10}^S = 3\} \quad \text{---(116)}$$

where v_l^S is the number of times the l^{th} *sensor information vector*, $\tilde{\mathbf{x}}_l^S$, appears in the combination, and all values v_l^S that are not listed in (114), (115) or (116) are zero.

The relevant sensor information are those referenced in (114), (115) and (116) that form the set:

$$\mathbf{x}^S = \{\tilde{\mathbf{x}}_6^S, \tilde{\mathbf{x}}_7^S, \tilde{\mathbf{x}}_8^S, \tilde{\mathbf{x}}_9^S, \tilde{\mathbf{x}}_{10}^S\} \quad \text{---(117)}$$

These relevant *sensor information vectors* are listed in Table 10.

| \mathbf{x}^S | | | |
|----------------|-----------------------------|---------|---------|
| l | $\tilde{\mathbf{x}}_l^S$ | x_1^S | x_2^S |
| 6 | $\tilde{\mathbf{x}}_6^S$ | 2 | 1 |
| 7 | $\tilde{\mathbf{x}}_7^S$ | 3 | 1 |
| 8 | $\tilde{\mathbf{x}}_8^S$ | 2 | 2 |
| 9 | $\tilde{\mathbf{x}}_9^S$ | 3 | 2 |
| 10 | $\tilde{\mathbf{x}}_{10}^S$ | 3 | 3 |

Table 10: List of relevant *sensor information vectors* for the system in Figure 30 with evidence in equations (95), (96) and (97).

Structure functions calculate the state of the sensors located at the (sub-) system levels based on

the *state vector* and system logic (such as that illustrated in Figure 29).

$$x_i^S = \phi_i^S(\tilde{\mathbf{x}}) = \phi_i^S(\{x_1, x_2, x_3, \dots, x_b, \dots, x_{n'}\})$$

$$= \begin{cases} z - 1 & \text{if the (sub-) system has completely failed} \\ \dots & \dots \\ 1 & \text{if the (sub-) system is in the first degraded state} \\ 0 & \text{if the (sub-) system is completely functional} \end{cases} \quad \text{---(55)}$$

Therefore, the *sensor information vector* can be defined as:

$$\tilde{\mathbf{x}}^S = \begin{Bmatrix} x_1^S \\ x_2^S \end{Bmatrix} = \begin{Bmatrix} \phi_1^S(\tilde{\mathbf{x}}) \\ \phi_2^S(\tilde{\mathbf{x}}) \end{Bmatrix} = \begin{Bmatrix} \phi_1^S(\{x_1, x_2, x_3, \dots, x_b, \dots, x_{n'}\}) \\ \phi_2^S(\{x_1, x_2, x_3, \dots, x_b, \dots, x_{n'}\}) \end{Bmatrix} \quad \text{---(118)}$$

The ***total criteria*** for *component state vectors*, $\tilde{\mathbf{x}} = \{x_1, x_2, x_3, \dots, x_b, \dots, x_{n'}\}$, is that they must imply a *sensor information vector* that is listed in the set (117) and Table 10.

$$\text{i.e. } \tilde{\mathbf{x}}^S = \begin{Bmatrix} x_1^S \\ x_2^S \end{Bmatrix} = \begin{Bmatrix} \phi_1^S(\tilde{\mathbf{x}}) \\ \phi_2^S(\tilde{\mathbf{x}}) \end{Bmatrix} = \begin{Bmatrix} \phi_1^S(\{x_1, x_2, x_3, \dots, x_b, \dots, x_{n'}\}) \\ \phi_2^S(\{x_1, x_2, x_3, \dots, x_b, \dots, x_{n'}\}) \end{Bmatrix} \in \mathbf{X}^S \quad \text{---(119)}$$

The ***partial criteria*** deal with *component state vector sub-sets*, $\{x_1, x_2, x_3, \dots, x_b\}$ where $b < n'$.

If $\{\min_1, \min_2, \dots, \min_b, \dots, \min_{n'}\}$ and $\{\max_1, \max_2, \dots, \max_b, \dots, \max_{n'}\}$ are the set of minimum and maximum possible state variable values, x_b , the ***partial criteria*** can be written as:

$$\tilde{\mathbf{x}}^{S(-)} = \begin{Bmatrix} x_1^{S(-)} \\ x_2^{S(-)} \end{Bmatrix} = \begin{Bmatrix} \phi_1^S(\tilde{\mathbf{x}}^{(-)}) \\ \phi_2^S(\tilde{\mathbf{x}}^{(-)}) \end{Bmatrix} = \begin{Bmatrix} \phi_1^S(\{x_1, \dots, x_b, \min_{b+1}, \dots, \min_{n'}\}) \\ \phi_2^S(\{x_1, \dots, x_b, \min_{b+1}, \dots, \min_{n'}\}) \end{Bmatrix}$$

$$\leq \begin{Bmatrix} x_1^S \\ x_2^S \end{Bmatrix} = \tilde{\mathbf{x}}^S, \tilde{\mathbf{x}}^S \in \mathbf{x}^S \quad \text{---(120)}$$

and $\tilde{\mathbf{x}}^{S(+)} = \begin{Bmatrix} x_1^{S(+)} \\ x_2^{S(+)} \end{Bmatrix} = \begin{Bmatrix} \phi_1^S(\tilde{\mathbf{x}}^{(+)}) \\ \phi_2^S(\tilde{\mathbf{x}}^{(+)}) \end{Bmatrix} = \begin{Bmatrix} \phi_1^S(\{x_1, \dots, x_b, \max_{b+1}, \dots, \max_{n'}\}) \\ \phi_2^S(\{x_1, \dots, x_b, \max_{b+1}, \dots, \max_{n'}\}) \end{Bmatrix}$

$$\geq \begin{Bmatrix} x_1^S \\ x_2^S \end{Bmatrix} = \tilde{\mathbf{x}}^S, \tilde{\mathbf{x}}^S \in \mathbf{x}^S \quad \text{---(121)}$$

The sets $\{\min_1, \min_2\}$ and $\{\max_1, \max_2\}$ are **trivially** set at:

$$\{\min_1, \min_2\} = \{0, 0\} \quad \text{---(122)}$$

$$\{\max_1, \max_2\} = \{(z-1), (z-1)\} = \{3, 3\} \quad \text{---(123)}$$

The minimum and maximum sets are improved based on iteratively undertaking (112) and (113) until the sets in (122) and (123) reach steady state shown in (124) and (125).

$$\{\min_1, \min_2\} = \{1, 0\} \quad \text{---(124)}$$

$$\{\max_1, \max_2\} = \{3, 3\} \quad \text{---(125)}$$

The **generation of cut-sets** is then carried out using the **partial criteria** represented in equations (120) and (121) and **total criterion** represented in equation (119). The process is illustrated in Figure 35.

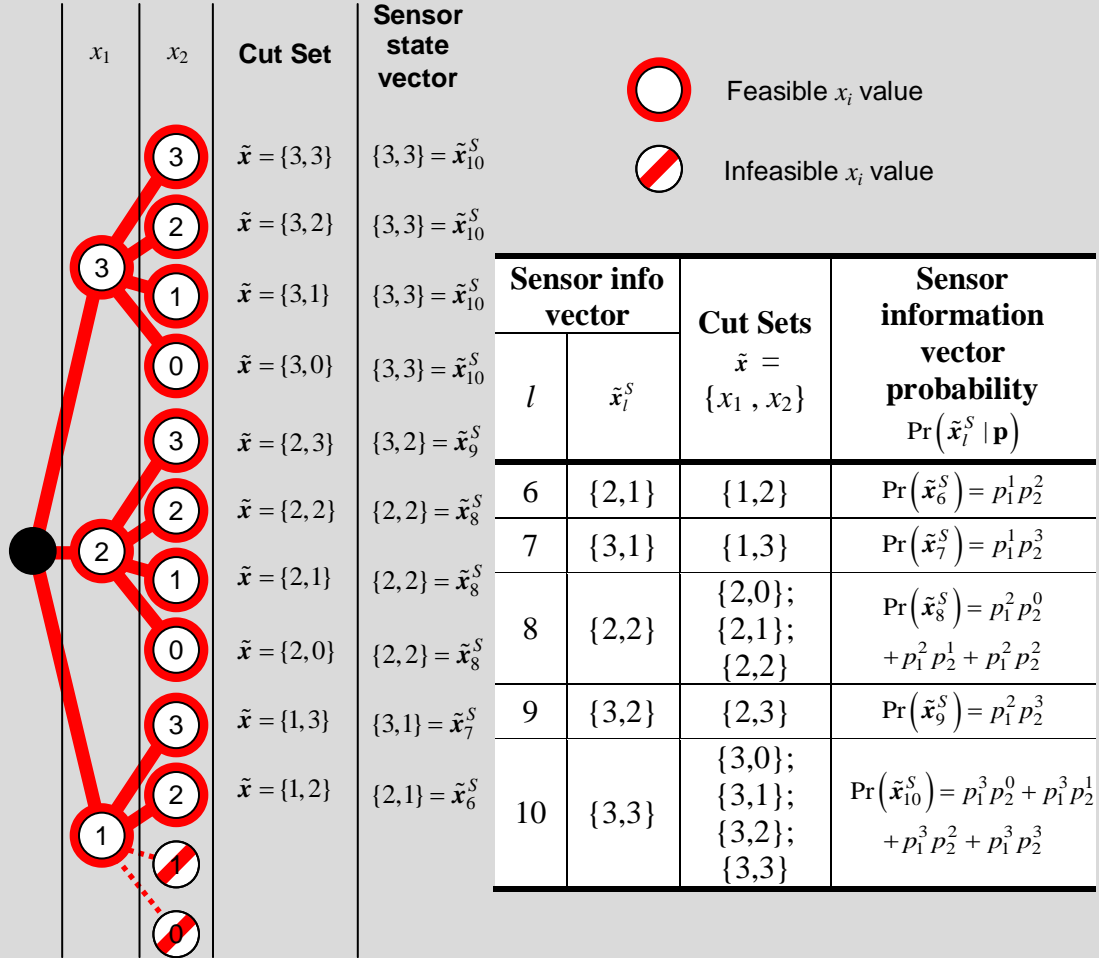


Figure 35: Generation of *component cut-sets*.

It can be seen that there are ten *cut-sets* that generate *sensor information vectors* that form *combinations* previously identified. The algorithm reduces the number of potential cut-sets that need to be considered from a total of $z^{n'} = 4^2 = 16$ to 12. This is not a large reduction due to the

high proportion of relevant cut-sets (10 out of 16), but this process will yield significant computational reductions in larger, more complex systems.

5.6. EVALUATING DOWNWARDS INFERENCE LIKELIHOOD FUNCTION

Recall that the likelihood function for downwards inference was modified to:

$$L(E | \mathbf{p}) \propto \sum_{\forall \tilde{\mathbf{v}}_a^S \in \mathbf{V}_E^S} \left[\prod_{l=1}^{l_{\max}} \left\{ \frac{\left[\Pr(\tilde{\mathbf{x}}_l^S | \mathbf{p}) \right]^{(v_l^S)_a}}{(v_l^S)_a!} \right\} \right] \quad \text{---(70)}$$

The outputs of the *combination generation* and *cut-set generation algorithms* allow equation (70) to be solved.

Example 16: Multi-state on-demand system likelihood function evaluation

Consider the two component series system in Example 14. The three possible *combinations of sensor information vectors* are:

$$\tilde{\mathbf{v}}_1^S = \{v_6^S = 1, v_7^S = 2, v_8^S = 4, v_{10}^S = 3\} \quad \text{---(114)}$$

$$\tilde{\mathbf{v}}_2^S = \{v_6^S = 2, v_7^S = 1, v_8^S = 3, v_9^S = 1, v_{10}^S = 3\} \quad \text{---(115)}$$

$$\tilde{\mathbf{v}}_3^S = \{v_6^S = 3, v_8^S = 2, v_9^S = 2, v_{10}^S = 3\} \quad \text{---(116)}$$

where v_l^S is the number of times the l^{th} sensor information vector, $\tilde{\mathbf{x}}_l^S$, appears in the

combination, and all values v_l^S that are not listed in (114), (115) or (116) are zero.

The *relevant sensor information vectors* form the set:

$$\mathbf{x}^S = \{\tilde{\mathbf{x}}_6^S, \tilde{\mathbf{x}}_7^S, \tilde{\mathbf{x}}_8^S, \tilde{\mathbf{x}}_9^S, \tilde{\mathbf{x}}_{10}^S\} \quad \text{---(117)}$$

The vectors in (117), their cut-sets and conditional probabilities are listed in Table 11.

| <i>Sensor information vector</i> | | Cut Sets $\tilde{\mathbf{x}} = \{x_1, x_2\}$ | $\Pr(\tilde{\mathbf{x}}_l^S \mathbf{p})$ |
|----------------------------------|--------------------------|--|--|
| l | $\tilde{\mathbf{x}}_l^S$ | | |
| 6 | {2,1} | {1,2} | $\Pr(\tilde{\mathbf{x}}_6^S) = p_1^{(1)} p_2^{(2)}$ |
| 7 | {3,1} | {1,3} | $\Pr(\tilde{\mathbf{x}}_7^S) = p_1^{(1)} p_2^{(3)}$ |
| 8 | {2,2} | {2,0}; {2,1}; {2,2} | $\Pr(\tilde{\mathbf{x}}_8^S) = p_1^{(2)} p_2^{(0)} + p_1^{(2)} p_2^{(1)} + p_1^{(2)} p_2^{(2)}$ |
| 9 | {3,2} | {2,3} | $\Pr(\tilde{\mathbf{x}}_9^S) = p_1^{(2)} p_2^{(3)}$ |
| 10 | {3,3} | {3,0}; {3,1}; {3,2}; {3,3} | $\Pr(\tilde{\mathbf{x}}_{10}^S) = p_1^{(3)} p_2^{(0)} + p_1^{(3)} p_2^{(1)} + p_1^{(3)} p_2^{(2)} + p_1^{(3)} p_2^{(3)}$ |

Table 11: List of relevant *sensor information vectors* for the system in Figure 30 with evidence in equations (95), (96) and (97) and mutually exclusive cut-sets.

Substituting these outputs into equation (70) yields:

$$\begin{aligned}
L(E|\mathbf{p}) &\propto \sum_{\forall \mathbf{v}_a^S \in \mathbf{V}_E^S} \left[\prod_{l=1}^{l_{\max}} \left\{ \frac{\left[\Pr(\tilde{\mathbf{x}}_l^S | \mathbf{p}) \right]^{(v_l^S)_a}}{(v_l^S)_a!} \right\} \right] \\
&= \left\{ \begin{aligned} &\frac{\Pr(\tilde{\mathbf{x}}_6^S | \mathbf{p})^1}{1!} \frac{\Pr(\tilde{\mathbf{x}}_7^S | \mathbf{p})^2}{2!} \frac{\Pr(\tilde{\mathbf{x}}_8^S | \mathbf{p})^4}{4!} \frac{\Pr(\tilde{\mathbf{x}}_{10}^S | \mathbf{p})^3}{3!} \\ &+ \frac{\Pr(\tilde{\mathbf{x}}_6^S | \mathbf{p})^2}{2!} \frac{\Pr(\tilde{\mathbf{x}}_7^S | \mathbf{p})^1}{1!} \frac{\Pr(\tilde{\mathbf{x}}_8^S | \mathbf{p})^3}{3!} \frac{\Pr(\tilde{\mathbf{x}}_9^S | \mathbf{p})^1}{1!} \frac{\Pr(\tilde{\mathbf{x}}_{10}^S | \mathbf{p})^3}{3!} \\ &+ \frac{\Pr(\tilde{\mathbf{x}}_6^S | \mathbf{p})^3}{3!} \frac{\Pr(\tilde{\mathbf{x}}_8^S | \mathbf{p})^2}{2!} \frac{\Pr(\tilde{\mathbf{x}}_9^S | \mathbf{p})^2}{2!} \frac{\Pr(\tilde{\mathbf{x}}_{10}^S | \mathbf{p})^3}{3!} \end{aligned} \right\} \\
&= \frac{1}{288} \left\{ \begin{aligned} &\frac{\Pr(\tilde{\mathbf{x}}_6^S | \mathbf{p})}{\times \Pr(\tilde{\mathbf{x}}_8^S | \mathbf{p})^2} \left[\begin{aligned} &\Pr(\tilde{\mathbf{x}}_7^S | \mathbf{p})^2 \Pr(\tilde{\mathbf{x}}_8^S | \mathbf{p})^2 \\ &+ 4 \Pr(\tilde{\mathbf{x}}_6^S | \mathbf{p}) \Pr(\tilde{\mathbf{x}}_7^S | \mathbf{p}) \Pr(\tilde{\mathbf{x}}_8^S | \mathbf{p}) \Pr(\tilde{\mathbf{x}}_9^S | \mathbf{p}) \\ &+ 2 \Pr(\tilde{\mathbf{x}}_6^S | \mathbf{p})^2 \Pr(\tilde{\mathbf{x}}_9^S | \mathbf{p})^2 \end{aligned} \right] \\ &\times \Pr(\tilde{\mathbf{x}}_{10}^S | \mathbf{p})^3 \end{aligned} \right\} \\
&= \frac{1}{288} \left\{ \begin{aligned} &(p_1^{(1)})^3 (p_1^{(2)})^4 (p_1^{(3)})^3 p_2^{(2)} (p_2^{(3)})^2 \\ &\times (p_2^{(0)} + p_1^{(2)} + p_2^{(2)})^2 (p_2^{(0)} + p_2^{(1)} + p_2^{(2)} + p_2^{(3)})^3 \\ &\times \left[\begin{aligned} &(p_2^{(0)} + p_1^{(2)} + p_2^{(2)})^2 + 2(p_2^{(2)})^2 \\ &+ 4p_2^{(3)}(p_2^{(0)} + p_1^{(2)} + p_2^{(2)}) \end{aligned} \right] \end{aligned} \right\} \quad \text{---(126)}
\end{aligned}$$

5.7. SUMMARY

Chapter 5 outlined an algorithm that is used to generate likelihood functions that allow Bayesian analysis of *overlapping data* sets from on-demand systems. At the heart of the algorithm is the development of sets of combinations of *sensor information vectors* that imply the evidence. A *sensor information vector* summarises the states detected by all sensors in a particular demand. Based on the number of demands, generating combinations of *sensor information vectors* that imply the evidence is the most computationally burdensome activity. By replacing component state vectors with *sensor information vectors*, the total number of possible vectors each

combination can include is considerably less (noting that there will be fewer sensors than components). The probability of each *sensor information vector* can be calculated using cut sets derived from the algorithm, and the likelihood function for specific evidence sets easily evaluated.

Chapter 6: *Overlapping data* likelihood function of systems with continuous life metrics

6.1. INTRODUCTION

In the case of systems based on time (which is a continuous life metric), the probability of the j^{th} component having failed at a given time t is defined by the set of reliability parameters of that component. This requires a fundamental change in approach to data analysis when compared to that for on-demand. This chapter outlines a methodology that allows Bayesian analysis of time-based systems that have *overlapping data* sets. The methodology is completely translatable to systems based on other continuous life metrics (such as distance).

6.2. TIME BASED FAILURE PROBABILITY

For a system, component failure probability is equivalent to the time based cumulative distribution function or *CDF*, $F(t)$. The failure probability then becomes a function of time. The *CDF* is defined by a set of parameters, which for the j^{th} component is represented as $\tilde{\theta}_j$. The set of all n component parameters for the system is:

$$\boldsymbol{\theta} = \{\tilde{\theta}_1, \tilde{\theta}_2, \tilde{\theta}_3, \dots, \tilde{\theta}_j, \dots, \tilde{\theta}_n\} \quad \text{---(127)}$$

As the *CDF* is the failure probability of each component at time t :

$$p_j(t|\tilde{\theta}_j) = F_j(t|\tilde{\theta}_j) = p_j|_{time=t} = p_j \quad \text{---(128)}$$

The probability that the j^{th} component will fail at time t_j is:

$$\Pr(T_j = t_j|\tilde{\theta}_j) = f_j(t_j|\tilde{\theta}_j)dt = dt \left[\frac{d}{dt} p_j(t|\tilde{\theta}_j) \right]_{t=t_j} = dt \left[\frac{d}{dt} F_j(t|\tilde{\theta}_j) \right]_{t=t_j} \quad \text{---(129)}$$

where $f_j(t_j|\tilde{\theta}_j)$ is the probability density function or *PDF* of the time to failure of the j^{th} component given the set of parameters $\tilde{\theta}_j$.

It is typical for *PDFs* to be used in likelihood functions dealing with continuous random variables, even though they are not by definition probabilities. *PDFs* are non-zero and exploit proportionality, whilst specific probabilities of a continuous random variable being a particular value are zero. Notwithstanding, expressing the probability of observing a continuous random variable as a factor of dt (even though this is as discussed theoretically equivalent to zero) is important for subsequent steps in the methodology proposed.

Recall that if *overlapping data sets* are denoted E^\bullet (each E^\bullet is overlapping with every other E^\bullet), then *overlapping data* within a system based on continuous life metrics is formally written in Bayes' theorem in equation (130).

$$\pi_1(\boldsymbol{\theta} | E = \{E_1^\bullet, E_2^\bullet, E_3^\bullet, \dots\}) = \frac{L(\{E_1^\bullet, E_2^\bullet, E_3^\bullet, \dots\} | \boldsymbol{\theta}) \pi_0(\boldsymbol{\theta})}{\int_{\forall \boldsymbol{\theta}'} L(\{E_1^\bullet, E_2^\bullet, E_3^\bullet, \dots\} | \boldsymbol{\theta}') \pi_0(\boldsymbol{\theta}') d\boldsymbol{\theta}'} \quad \text{---(130)}$$

where $\{E_1^\bullet, E_2^\bullet, E_3^\bullet, \dots\}$ is a number of *overlapping data*/evidence sub-sets of the total evidence E .

As *overlapping data sets* are dependent, the overall likelihood function is not a product of individual data set likelihood functions.

$$L(E = \{E_1^\bullet, E_2^\bullet, E_3^\bullet, \dots\} | \boldsymbol{\theta}) \neq \prod_{\forall i} L(E_i^\bullet | \boldsymbol{\theta}) \quad \text{---(131)}$$

The inherent dependence of *overlapping data* means that each set is dependent on some or all of the other sets, and the likelihood function in equation (131) can be amended:

$$L(E = \{E_1^\bullet, E_2^\bullet, E_3^\bullet, \dots\} | \boldsymbol{\theta}) = \prod_{\forall i} L(E_i^\bullet | \boldsymbol{\theta}, \subseteq \forall E_{\neq i}^\bullet) \quad \text{---(132)}$$

As studied previously, it is often the case where sensors are placed at various hierarchical levels of systems. Evidence is then taken as the time failure is detected by various sensors and is by definition *overlapping*.

$$E = \tilde{t}^S = \{t_1^S, t_2^S, t_3^S, \dots, t_i^S, \dots, t_m^S\} \quad \text{---(133)}$$

where E is the evidence set of the complex system, m is the number of sensors in the system, and t_i^S is the time failure is detected by the i^{th} sensor.

To better understand the nature of *overlapping data sets* in the context of time-based systems, the concept of *inference diagrams* is introduced. Each sub-set is defined by an *inference diagram*. In effect, *inference diagrams* modify the likelihood function in equation (132) by generating separate likelihood functions for each sub-set of the evidence, and generating an overall likelihood function through their product.

Example 17: Inference Diagrams

Consider the 4 component, 3 sensor system in Figure 36. The different colours denote different regions of inference and influence for each sensor. Components 2 and 3 are identical components (i.e. components from the same *component type*). The *component type number* of the j^{th} component, j_b , is listed in Table 12.

| Component Number $b = \in (1, 2, \dots, n)$ | Component Type $j_b \in (1, 2, \dots, n)$ | Component Failure Probability p_{j_b} |
|--|--|---|
| 1 | 1 (type A) | p_1 |
| 2 | 2 (type B) | p_2 |
| 3 | 2 (type B) | p_2 |
| 4 | 3 (type C) | p_3 |

Table 12: *Component numbers and component type numbers for components in the system illustrated in figure 16.*

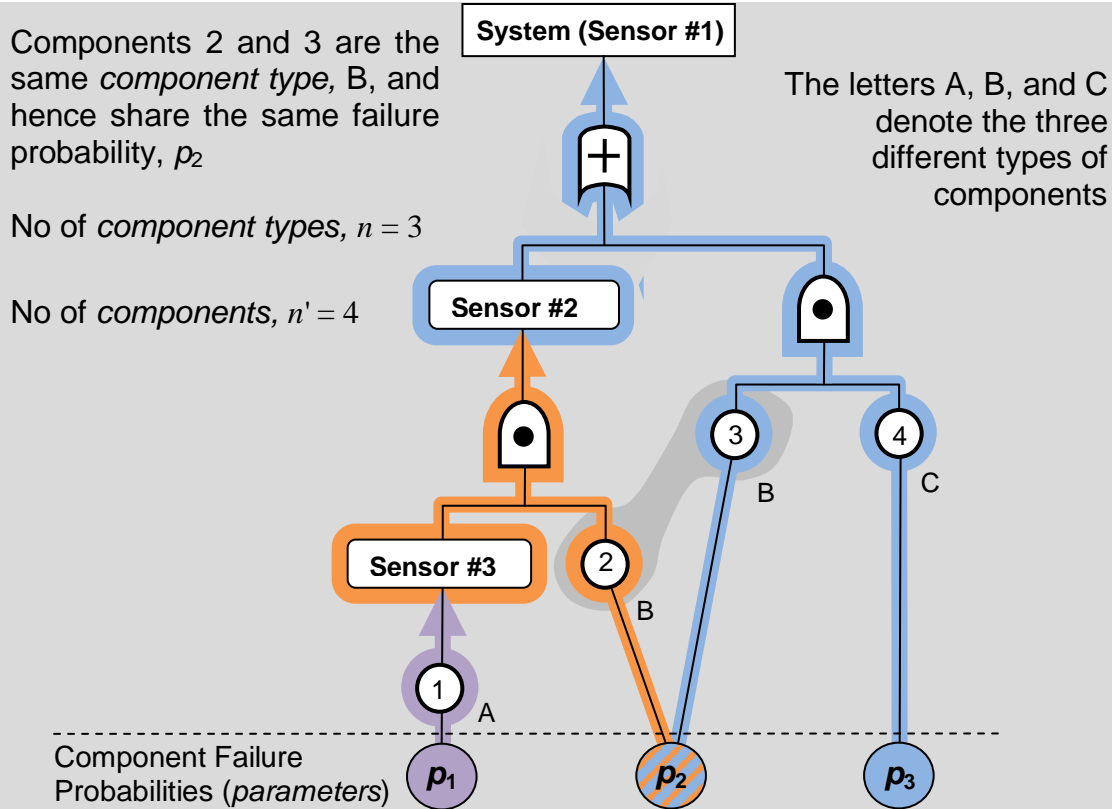
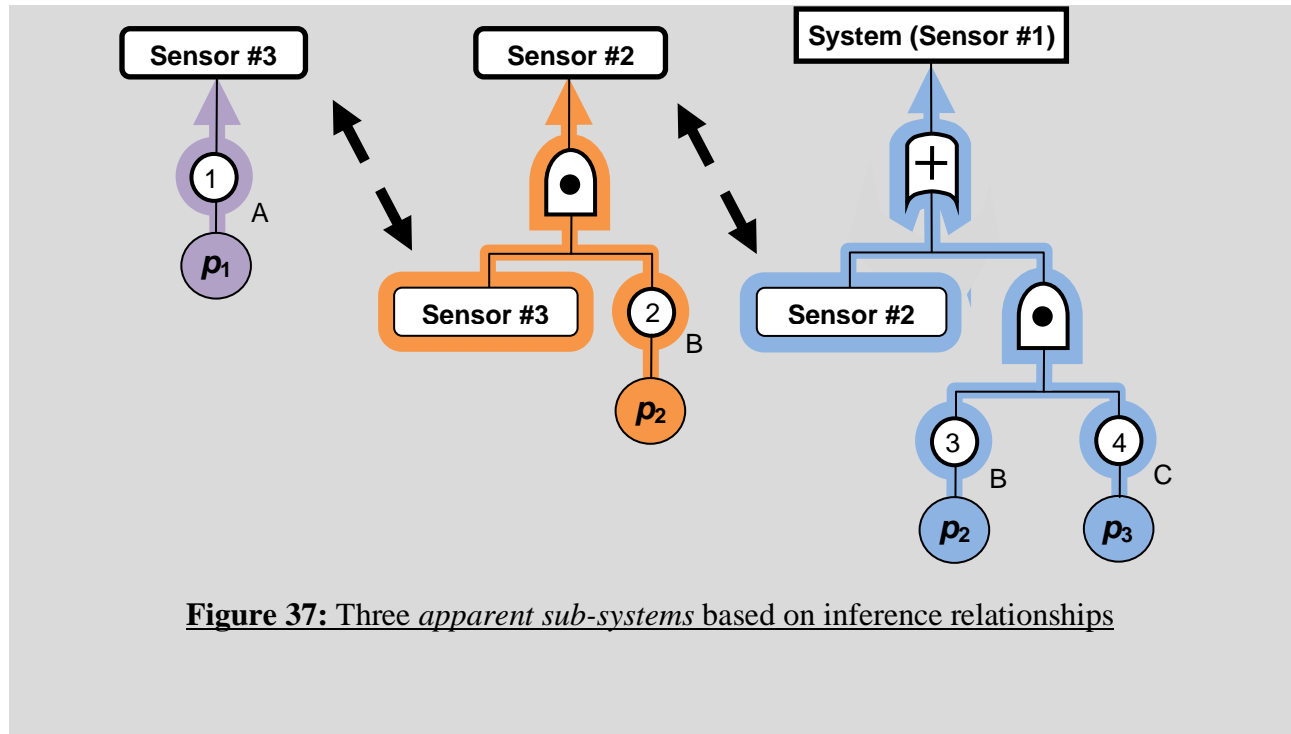


Figure 36: Three sensor on-demand system with *paths of apparent influence* indicated

It can be seen that there are three distinct *apparent inference paths*. The blue inference path terminates at the top event or system level, and shows that components 1, 2 and sensor #3 data does not have an *apparent influence* on system level characteristics since sensor #2 summarizes all subordinate structural characteristics. Should sensor #3 evidence change or be re-evaluated, it will have no implication on system level evidence if the sensor #2 evidence remains unchanged. In this way, sensors can ‘isolate’ the inference and influence of subordinate systems, sub-systems and components. The inference of each sensor separate the system in figure 16 into three separate sub-systems as illustrated in Figure 37.



The inference diagram introduced in Example 17 is a *directed acyclic graph* that illustrates paths of apparent inference and how sensor data summarizes the effect that subordinate components have on superior sensors. *Inference diagrams* are technically a form of an *influence diagram* primarily emanating from the fields of Bayesian networks and decision-making methodologies. Nevertheless, some distinction between the two is worthwhile as *inference diagrams* only involve ‘events’ in the form of component functionality, and are meant to assign inference from components to sensors rather than being a fundamental part of calculation. They also include additional information on the nature of dependence between components by the inclusion of logical gates (such as the ‘AND’ / ‘OR’ gates of a fault-tree). The colour shading in Figure 36 and Figure 37 emphasizes how the system is broken into sub-systems, while the grey shading highlights that components 2 and 3 both share failure probability p_2 . [18]

In this way, the apparent inference from superior sensor data comes from subordinate sensor data and subordinate components. For example, components 3 and 4 in Figure 36 and Figure 37 are subject to inference from sensors #1 and #2 only; how components 1 and 2 behave does not matter given data from sensor #2 is gathered (hence the term apparent inference).

The system in Figure 36 is equally well represented by the three separate sub-systems in Figure 37. By definition, they are *overlapping* through the dependency between failures detected by the same sensor represented in different sub-systems. However, the sub-systems can be treated as *non-overlapping* when each is calculated in the context of all other evidence sets (which would be the product of individual likelihood functions of each sub-system). As components 2 and 3 are identical but appear in different sub-systems, the likelihood of the component type failure probability is influenced by two of the three sub-system likelihood functions.

As illustrated in Figure 38 (which replicates the system represented in Figure 36 and Figure 37), the inference from any sub-set of the evidence must be considered in the context provided by all other evidence. For example, sensor #1 will always detect failure at the same time sensor #2 detects failure if component 2 has not previously failed.

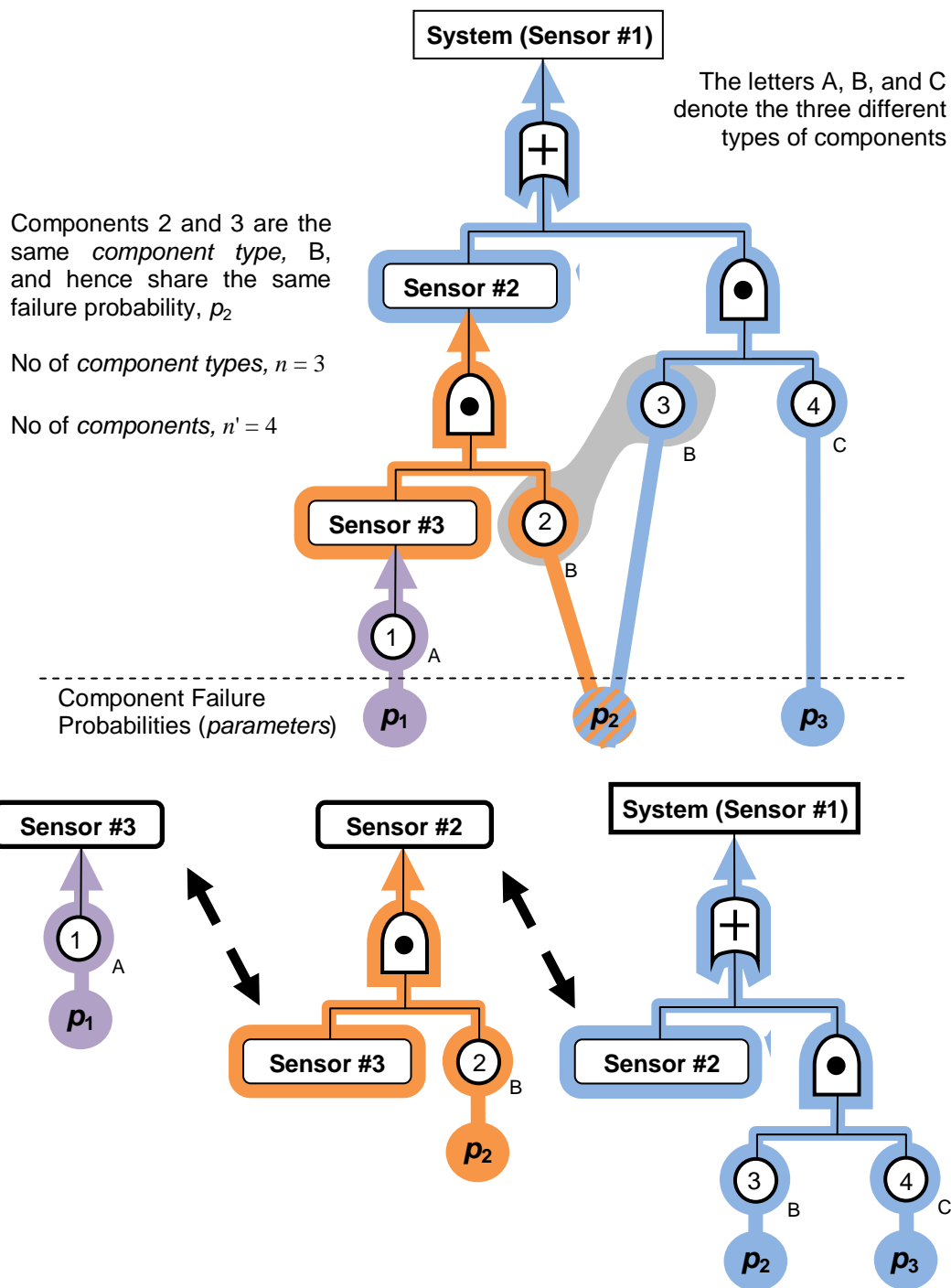


Figure 38: Three *apparent sub-systems* based on inference relationships

6.3. HIERARCHY OF SENSORS AND COMPONENTS

If a system has m sensors, then the hierarchy of each sensor needs to be understood. The following definition and terminology is based on any reliability graphical representation of a system but is most easily observed within a fault-tree. Any component or sensor that is hierarchically beneath a particular point but not necessarily within the same branch in a graphical representation of a system is said to be at a lower-level to that point. Conversely, any component or sensor that is hierarchically above a particular point but not necessarily within the same branch is said to be at a higher-level to that point. Any component or sensor that is at the same level hierarchically is said to be equivalent.

Any lower-level component or sensor that appears within the same branch of a particular point is said to be subordinate. Likewise, any higher-level component or sensor that appears within the same branch of a particular point is said to be superior. For any sensor, all subordinate components and sensors that appear in its relevant *inference diagram* (as per the example in Figure 38) are said to be inferentially subordinate.

Let the 1st sensor be such be the ‘highest’ sensor and the m^{th} sensor be the ‘lowest’. This means that in any fault-tree representation, the $(i+1)^{\text{th}}$ sensor will always be represented below or hierarchically equivalent to the i^{th} sensor. *Inferentially subordinate* sensors to the i^{th} sensor must be drawn from the set of the $(i+1)^{\text{th}}$ to m^{th} sensors. This ordering is apparent in Figure 39.

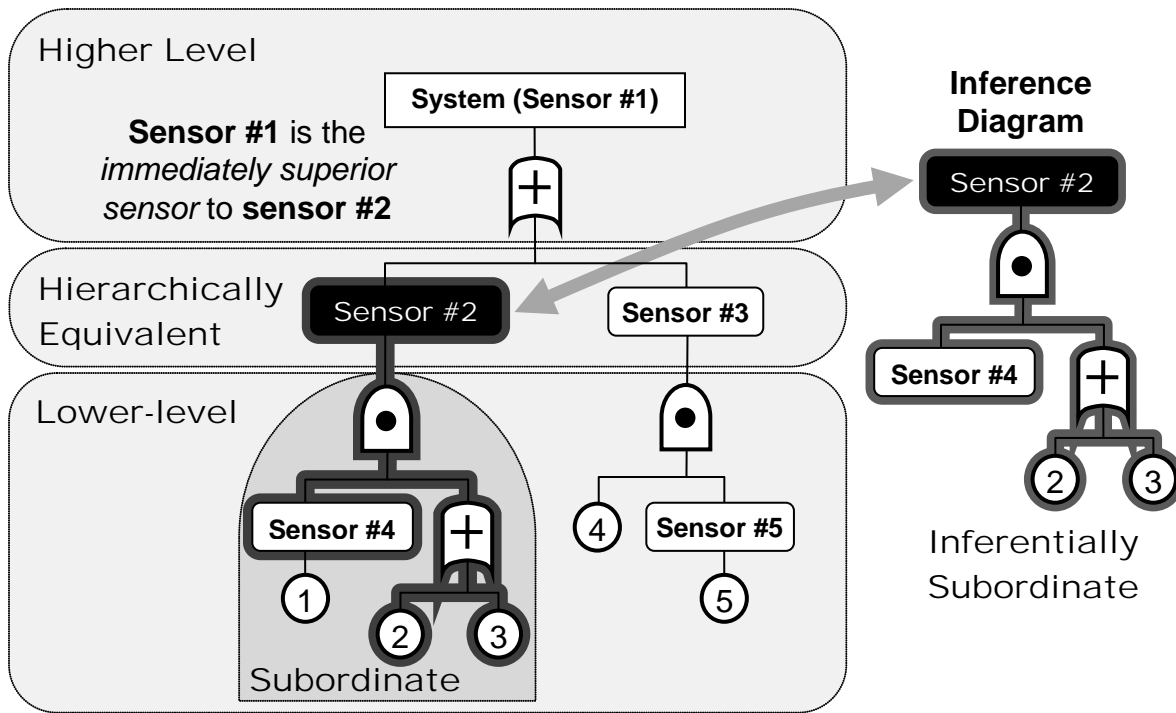


Figure 39: Hierarchy of a 5 component, 5 sensor system *with respect to sensor #2*

Figure 38 also illustrates how data from subordinate sensors effectively summarizes the effect that lower level components have on higher level sensors. Therefore, the *time to failure detection* characteristics of each sensor is conditional on *inferentially subordinate* components and sensors only. Referring to Figure 39, even though component 1 is *subordinate* to sensor #2, it is not *inferentially subordinate* to sensor #2 as sensor #4 provides all relevant information about the state of the system at that point. Given that sensor #4 exists, the behaviour of component 1 is no longer relevant. This allows the generation of *inferentially subordinate* sets for each sensor as listed in Table 13.

| Sensor Number i | <i>Inferentially subordinate sets</i> | | | | | | |
|----------------------|---|--------------------------------------|---|--|---|--|--|
| | sensor indices $\tilde{i}_i^{\subset S}$ | sensors $\tilde{S}_i^{\subset S}$ | sensor detection times $\tilde{t}_i^{\subset S}$ | component indices $\tilde{j}_i^{\subset S}$ | components $\tilde{C}_i^{\subset S}$ | component failure probabilities $\tilde{p}_i^{\subset S}$ | component reliability parameters $\tilde{\theta}_i^{\subset S}$ |
| 1 | {2,3} | { S_2, S_3 } | { t_2^S, t_3^S } | \emptyset | \emptyset | \emptyset | \emptyset |
| 2 | {4} | { S_4 } | { t_4^S } | {2,3} | { C_2, C_3 } | { p_2, p_3 } | { $\tilde{\theta}_2, \tilde{\theta}_3$ } |
| 3 | {5} | { S_5 } | { t_5^S } | {4} | { C_4 } | { p_4 } | { $\tilde{\theta}_4$ } |
| 4 | \emptyset | \emptyset | \emptyset | {1} | { C_1 } | { p_1 } | { $\tilde{\theta}_1$ } |
| 5 | \emptyset | \emptyset | \emptyset | {5} | { C_5 } | { p_5 } | { $\tilde{\theta}_5$ } |

Table 13: Expression of multi-level evidence for binary-state on-demand systems

Inferentially subordinate sets are formally defined as:

$\tilde{i}_i^{\subset S}$... the set of indices of all sensors that are *inferentially subordinate* to the i^{th} sensor;

$\tilde{S}_i^{\subset S}$... the set of all sensors that are *inferentially subordinate* to the i^{th} sensor;

$\tilde{t}_i^{\subset S}$... the set of all failure detection times of sensors that are *inferentially subordinate* to the i^{th} sensor (i.e. the failure detection times of all sensors $\in \mathbf{S}_i^{\subset S}$);

$\tilde{j}_i^{\subset S}$... the set of indices of all components that are *inferentially subordinate* to the i^{th} sensor;

$\tilde{C}_i^{\subset S}$... the set of all components that are *inferentially subordinate* to the i^{th} sensor;

$\tilde{p}_i^{\subset S}$... the set of failure probabilities of all components that are *inferentially subordinate* to the i^{th} sensor (i.e. the failure probabilities of all components $\in \mathbf{C}_i^{\subset S}$); and

$\tilde{\theta}_i^{\subset S}$... the set of reliability parameters of all components that are *inferentially subordinate* to the i^{th} sensor (i.e. the reliability parameters of all components $\in \mathbf{C}_i^{\subset S}$).

As all of the sets above are drawn from internally from the system, the following can also be written:

$$\tilde{p}_i^{\subseteq S} \subseteq \tilde{p} \text{ and } \boldsymbol{\theta}_i^{\subseteq S} \subseteq \boldsymbol{\theta} \quad \text{---(134)(135)}$$

Therefore, generalizing the likelihood function in equation (132) allows the posterior distribution using evidence taken as the times failure is detected by various sensors is:

$$\begin{aligned} \pi_1(\boldsymbol{\theta} | E) &= \pi_1\left(\boldsymbol{\theta} | \left\{E_1^\bullet, E_2^\bullet, E_3^\bullet, \dots, E_i^\bullet, \dots, E_m^\bullet\right\}\right) = \pi_1\left(\boldsymbol{\theta} | \left\{t_1^S, t_2^S, t_3^S, \dots, t_i^S, \dots, t_m^S\right\}\right) \\ &= \frac{\left[\prod_{i=1}^m L\left(t_i^S | \boldsymbol{\theta}_i^{\subseteq S}, \tilde{t}_i^{\subseteq S}\right)\right] \pi_0(\boldsymbol{\theta})}{\int_{\forall \boldsymbol{\theta}'} \left[\prod_{i=1}^m L\left(t_i^S | \boldsymbol{\theta}_i^{\subseteq S}, \tilde{t}_i^{\subseteq S}\right)\right] \pi_0(\boldsymbol{\theta}') d\boldsymbol{\theta}'} \end{aligned} \quad \text{---(136)}$$

where E is the evidence set $\{t_1^S, t_2^S, t_3^S, \dots, t_i^S, \dots, t_m^S\}$ and t_i^S is time failure is detected by the i^{th} sensor.

It can be seen that equation (136) is of similar construction to what one would expect when compiling multiple *non-overlapping data* sets (by simply multiplying individual likelihood functions). The key difference is that the likelihood function for each *overlapping data element* (which is the time to failure detection of the i^{th} sensor) is conditional on all inferentially subordinate failure detection times. This allows the dependence from the *overlapping* nature of the data to be incorporated into a Bayesian construct that can be more easily evaluated. The

likelihood function in equation (136) utilizes a *state of knowledge* dependence to incorporate multiple instances of the same component type. This dependence assumes that each component has the same underlying reliability parameters.

6.4. PROBABILITY DENSITY FUNCTIONS OF TIMES THAT SENSORS DETECT FAILURE

The development of a likelihood function for a system using evidence of the form introduced in equation (133) requires the time to failure detection probabilities of all sensors to be calculated. This involves the *PDF* of time to failure being calculated for the relevant points in the system. Using the system illustrated in Figure 38, the relationship between t^S (time sensor detects failure) and t (times that components fail) is illustrated in Figure 40.

Using the influence diagrams, the representation within Figure 38 can be represented as shown in Figure 41.

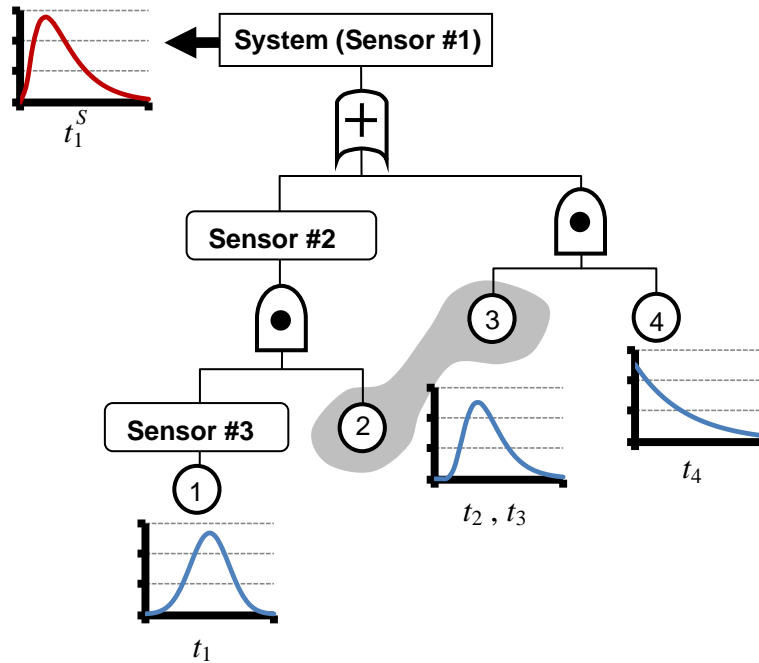


Figure 40: Representation of the probabilistic relationship between component failure times and times to detection of failure by sensors.

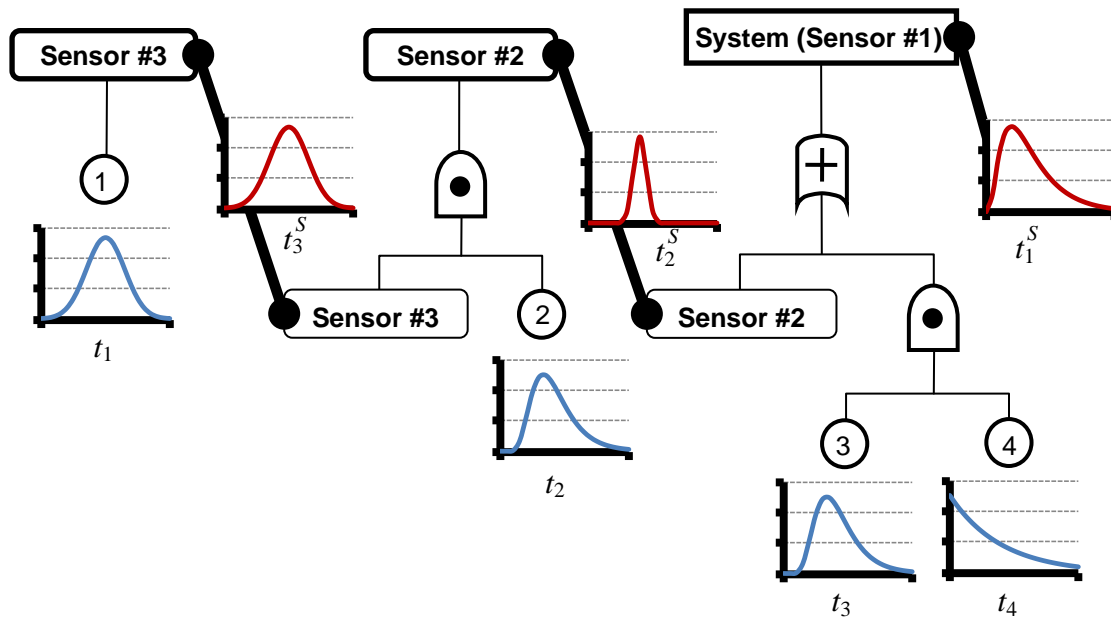


Figure 41: Representation of the probabilistic relationship between component failure times and sensor failure detection times – sensor inference sub-systems.

The *CDF* of the time to failure detection by the i^{th} sensor is based on the *CDF* of time to failure (detection) of *inferentially subordinate* components and sensors. The *PDF* is the derivative of the *CDF* with respect to t .

$$\text{i.e. probability of } i^{\text{th}} \text{ sensor failure detection } F_i^S(t | \boldsymbol{\theta}_i^{\subset S}, \tilde{\mathbf{t}}_i^{\subset S}) = p_i^S \Big|_t = p_i^S(\forall p_{i'}, \forall p_j) \Big|_t \text{---(137)}$$

where $i' \in \mathbf{i}_i^{\subset S}$ and $j \in \mathbf{j}_i^{\subset S}$.

All probabilities in equation (137) are functions of time and conditional on *inferentially subordinate* parameters, and equivalent to respective *CDFs* (i.e. $CDF \equiv p$). The *CDF* of the time to failure detection is a function based on system logic (through generation of disjoint cut-sets or equivalent method). The *PDF* of time to sensor detection by the i^{th} sensor is the derivative of equation (137) with respect to time.

$$\text{i.e. } f_i^S(t | \boldsymbol{\theta}_i^{\subset S}, \mathbf{t}_i^{\subset S}) = \frac{dp_i^S}{dt} = \frac{d}{dt} p_i^S(\forall p_{i'}, \forall p_j) = \left(\sum_{i' \in \mathbf{i}_i^{\subset S}} \frac{\partial p_i^S}{\partial p_{i'}} \bullet \frac{dp_{i'}^S}{dt} \right) + \left(\sum_{j \in \mathbf{j}_i^{\subset S}} \frac{\partial p_i^S}{\partial p_j} \bullet \frac{dp_j}{dt} \right) \text{---(138)}$$

where $i' \in \mathbf{i}_i^{\subset S}$ and $j \in \mathbf{j}_i^{\subset S}$.

When sensor evidence or data is introduced, the probabilistic relationships in Figure 40: and Figure 41 are broken between separate inference diagrams as previously discussed. This is shown in Figure 42 where all inferentially subordinate sensors have precisely known times to failure detection.

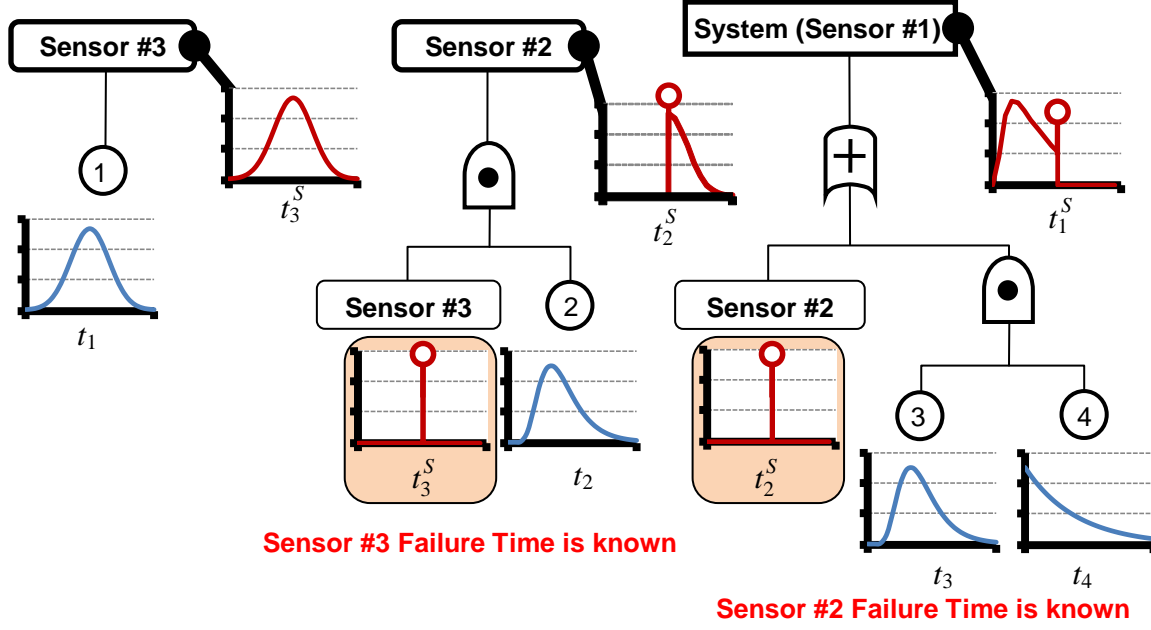


Figure 42: Representation of the probabilistic relationship between component failure times and times at which sensors detect failure – sensor inference sub-systems (where the times to failure detection of sensors #2 and #3 are known).

The evidence defines the characteristics of all *inferentially subordinate* sensors, which are those that appear on the right hand side of equations (137) and (138). The *CDF* of the time to failure detection of *inferentially subordinate* sensors becomes the *unit* or *Heaviside step function*⁵. This modifies equation (137) to:

$$F_i^S(t | \theta_i^{\subset S}, \tilde{t}_i^{\subset S}) = p_i^S = p_i^S(\forall p_i^S, \forall p_j) \quad \text{---(139)}$$

where $p_i^S = H(t - t_i^S)$ and $H(x)$ is the *unit* or *Heaviside step function*.

⁵ In this dissertation, the right continuous *Heaviside step function* will be used where $H(x) = 1$ when $x \geq 0$, 0 otherwise.

$$\begin{aligned}
f_i^S(t | \theta_i^{cS}, \tilde{t}_i^{cS}) &= \frac{dp_i^S}{dt} = \left(\sum_{i' \in \tilde{t}_i^{cS}} \frac{\partial p_{i'}^S}{\partial p_{i'}^S} \cdot \frac{dH(t - t_{i'}^S)}{dt} \right) + \left(\sum_{j \in \tilde{t}_i^{cS}} \frac{\partial p_j^S}{\partial p_j} \cdot \frac{dp_j}{dt} \right) \\
&= \left(\sum_{i' \in \tilde{t}_i^{cS}} \frac{\partial p_{i'}^S}{\partial p_{i'}^S} \cdot \delta(t - t_{i'}^S) \right) + \left(\sum_{j \in \tilde{t}_i^{cS}} \frac{\partial p_j^S}{\partial p_j} \cdot \frac{dp_j}{dt} \right) \quad \text{---(140)}
\end{aligned}$$

This allows the modification of the *CDF* of to time to failure detection by superior sensors, and introduces ‘steps’. For example, the *CDFs* for sensors #1 and #2 in Figure 42 are illustrated in Figure 43. It can be seen how the failure detection by subordinate sensors introduces ‘steps’ into the *CDF* of the superior sensor time to failure detection. As the conditional *PDF* in equation (140) is in effect the derivative of these stepped functions, special care must be taken when dealing with them mathematically. The *PDFs* of the *CDFs* below are undefined at the times when subordinate sensors detect failure, and hence cannot be substituted directly into Bayes’ Theorem. In this context, the probability of observing a particular instance of a continuous random variable (which is typically zero), will actually be finite when it coincides with the failure detection times of subordinate sensors.

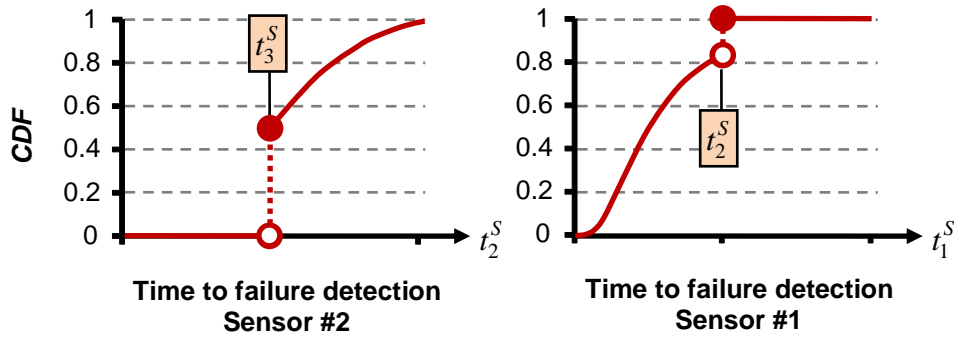


Figure 43: Conditional CDFs of time to failure detection of sensors #2 and #1 respectively,
given the failure detection time of subordinate sensors.

Throughout this dissertation, the following taxonomy will be adhered to:

- a. $CDF \left(F_i^S, p_i^S \right)$ and $PDF \left(f_i^S, \frac{dp_i^S}{dt} \right)$ apply to *the sensor being analysed*. These functions are conditional on *inferentially subordinate* sensors and components. The functions are directly utilized in the development of the **likelihood function**, as they calculate probabilities associated with the time the i^{th} sensor detects failure.
- b. *Evidential probability of failure detection* $\left(p_i^S \right)$ applies to *inferentially subordinate sensors*. This function is primarily conditional on observed times to failure, as they contextualize the functions in the previous sub-paragraph.

As equation (140) is the *PDF* of time to failure detection by the i^{th} sensor at time t , it is equivalent to the likelihood function:

$$\begin{aligned}
 \text{i.e. } L_i^S \left(t_i^S \mid \tilde{t}_i^{\subset S}, \boldsymbol{\theta}_i^{\subset S} \right) &= f_i^S \left(t_i^S \mid \boldsymbol{\theta}_i^{\subset S}, \tilde{t}_i^{\subset S} \right) \equiv \frac{dp_i^S}{dt} \Big|_{t=t_i^S} \\
 &= \left(\sum_{i' \in \tilde{t}_i^{\subset S}} \frac{\partial p_{i'}^S}{\partial p_{i'}^S} \bullet \frac{dH(t - t_{i'}^S)}{dt} \right)_{t=t_i^S} + \left(\sum_{j \in \tilde{j}_i^{\subset S}} \frac{\partial p_j^S}{\partial p_j^S} \bullet \frac{dp_j^S}{dt} \right)_{t=t_i^S} \\
 &= \left(\sum_{i' \in \tilde{t}_i^{\subset S}} \frac{\partial p_{i'}^S}{\partial p_{i'}^S} \bullet \frac{dH(t - t_{i'}^S)}{dt} \right)_{t=t_i^S} + \left(\sum_{j \in \tilde{j}_i^{\subset S}} \frac{\partial p_j^S}{\partial p_j^S} \Big|_{t=t_i^S} \bullet f_j(t_i^S \mid \tilde{\boldsymbol{\theta}}_j) \right) \quad \text{---(141)}
 \end{aligned}$$

where the *PDF* of time to failure of the j^{th} component, $f_j(t_i^S \mid \tilde{\boldsymbol{\theta}}_j) = \frac{dp_j^S}{dt} \Big|_{t=t_i^S}$.

The *Dirac delta function* is defined as 0 when $x \neq 0$ and $+\infty$ when $x = 0$ (and hence is not a proper function), making the evaluation of equation (141) problematic. Appendix A outlines how a likelihood function of the form in equation (141) is dealt with in a Bayesian context. Therefore, the likelihood function becomes:

$$L_i^S(t_i^S | \tilde{t}_i^{\subset S}, \theta_i^{\subset S}) \equiv \begin{cases} \sum_{i' \in \tilde{i}} \frac{\partial p_i^S}{\partial p_{i'}^S} \Big|_{t=t_i^S} & \dots \quad \tilde{i}_i \neq \emptyset \text{ and } \sum_{i' \in \tilde{i}} \frac{\partial p_i^S}{\partial p_{i'}^S} \neq 0 \\ \sum_{j \in \tilde{j}_i^{\subset S}} \frac{\partial p_i^S}{\partial p_j^S} \Big|_{t=t_i^S} \bullet f_j(t_i^S | \tilde{\theta}_j) & \dots \quad \text{otherwise} \end{cases} \quad \text{---(142)}$$

given $\tilde{i}_i = \{\forall i'\}$ such that $t_{i'}^S = t_i^S$ and $i' \in \tilde{i}_i^{\subset S}$.

Equation (142) will be a function of *inferentially subordinate* sensor failure detection probabilities. Recall that in each case, this is defined as:

$$p_{i'}^S(t | t_{i'}^S) = H(t - t_{i'}^S) = \begin{cases} 0 \dots t < t_{i'}^S \\ 1 \dots t \geq t_{i'}^S \end{cases} \quad \text{---(143)}$$

The likelihood of observing the set of failure detection times of the m sensors, $\tilde{t}^S = \{t_1^S, t_2^S, t_3^S, \dots, t_i^S, \dots, t_m^S\}$ given the set of parameters that define the reliability (and failure probability) characteristics of all system components $\theta = \{\tilde{\theta}_1, \tilde{\theta}_2, \tilde{\theta}_3, \dots, \tilde{\theta}_j, \dots, \tilde{\theta}_n\}$, is defined below. The likelihood functions derived from the data set of each sensor can now be multiplied as they have been effectively isolated into separate sub-systems of non-overlapping inference.

$$L(\tilde{\mathbf{t}} | \boldsymbol{\theta}) \propto L(\{t_1^S, \dots, t_m^S\} | \tilde{\mathbf{t}}_1, \dots, \tilde{\mathbf{t}}_n) = \prod_{i=1}^m L_i^S(t_i^S | \tilde{\mathbf{t}}_i^{\subset S}, \boldsymbol{\theta}_i^{\subset S})$$

$$\equiv \prod_{i=1}^m \left[\begin{array}{ll} \sum_{i' \in \tilde{\mathbf{t}}_i} \frac{\partial p_{i'}^S}{\partial p_i^S} \Big|_{t=t_i^S} & \dots \quad \tilde{\mathbf{t}}_i \neq \emptyset \text{ and } \sum_{i' \in \tilde{\mathbf{t}}_i} \frac{\partial p_{i'}^S}{\partial p_i^S} \neq 0 \\ \sum_{j \in \tilde{\mathbf{t}}_i^{\subset S}} \frac{\partial p_j^S}{\partial p_j} \Big|_{t=t_i^S} \bullet f_j(t_i^S | \tilde{\boldsymbol{\theta}}_j) & \dots \quad \text{otherwise} \end{array} \right] \quad \text{---(144)}$$

given $\tilde{\mathbf{t}}_i = \{\forall i'\}$ such that $t_{i'}^S = t_i^S$ and $i' \in \tilde{\mathbf{t}}_i^{\subset S}$ and where $p_{i'}^S|_t = H(t - t_{i'}^S)$,
 $p_j|_t = F_j(t | \tilde{\boldsymbol{\theta}}_j)$ and $\frac{dp_j}{dt} \Big|_t = f_j(t | \tilde{\boldsymbol{\theta}}_j)$.

6.5. CENSORED DATA

Thus far, *overlapping data* sets have been considered where all sensors, systems and sub-systems are allowed to operate until failure is detected. The evidence set, $\tilde{\mathbf{t}}^S = \{t_1^S, t_2^S, t_3^S, \dots, t_i^S, \dots, t_m^S\}$, contains specific times or inequalities for all t_i^S , which is the time that the i^{th} sensor detects failure. It is not unusual for systems to cease being operated (or observed) once sensor level failure occurs. In this instance, if t^* is the time that the relevant sub-system is ‘tested’, then it can be written:

$$t_i^S \text{ is } \begin{cases} = t & \dots & \text{where } t \leq t_i^* \\ \{\forall t > t_i^*\} & \dots & \text{(i.e. the set of all times } t \text{ greater than } t_i^*) \end{cases} \quad \text{---(145)}$$

If the i^{th} sensor has not detected failure by time t_i^* , (i.e. $t_i^S = \{\forall t > t_i^*\}$), then the *likelihood* of the corresponding observation is simply the complement of probability of failure detection by the i^{th} sensor by time t_i^* .

$$\text{i.e. } L_i^S(t_i^S | \tilde{t}_i^{\subset S}, \boldsymbol{\theta}_i^{\subset S}) = 1 - p_i^S \Big|_{t=t_i^*} \text{ if } t_i^S = \{\forall t > t_i^*\} \quad \text{---(146)}$$

All *inferentially subordinate* sensors (recalling that they are primarily conditional on observed failure detection time) need to be modified to incorporate right censored data. As *inferentially subordinate* sensor probability functions contextualize the likelihood functions of higher-level sensor data, a provision must be made for scenarios where *inferentially subordinate* sensors cease to be observed prior to higher-level sensors cease observation.

$$\text{i.e. } p_{i'}^S \Big|_t = \begin{cases} \frac{f_{i'}^S(t | \tilde{t}_{i'}^{\subset S}, \boldsymbol{\theta}_{i'}^{\subset S})}{1 - F_{i'}^S(t_{i'}^* | \tilde{t}_{i'}^{\subset S}, \boldsymbol{\theta}_{i'}^{\subset S})} & \dots \text{ if } t_{i'}^S = \{\forall t > t_{i'}^*\} \text{ and } t > t_{i'}^* \\ H(t - t_{i'}^S) & \dots \text{ otherwise} \end{cases} \quad \text{---(147)}$$

In effect, equation (147) allows scenarios where subordinate sensors are right censored, meaning that the only thing that is known about their respective sub-system is that eventual time to failure will occur sometime after $t_{i'}^*$. Therefore, the ‘state of knowledge’ of the eventual time to failure becomes a truncated *PDF* defined on $t > t_{i'}^*$.

This modifies the likelihood function derived in equation (144), which is based exclusively on an evidence set where the system is allowed to operate until all sensors detect failure, to the generalized form in equation (148).

$$\begin{aligned}
L(\tilde{\mathbf{t}} | \boldsymbol{\theta}) &= L(\{t_1^S, \dots, t_m^S\} | \tilde{\mathbf{t}}_1, \dots, \tilde{\mathbf{t}}_n) = \prod_{i=1}^m L_i^S(t_i^S | \tilde{\mathbf{t}}_i^{\subset S}, \boldsymbol{\theta}_i^{\subset S}) \\
&\equiv \prod_{i=1}^m \left[\begin{array}{ll} 1 - p_i^S \Big|_{t=t_i^*} & \dots \text{ if } t_i^S = \{\forall t > t_i^*\} \text{ (i.e. right censored data)} \\ \sum_{i' \in \tilde{\mathbf{t}}_i} \frac{\partial p_i^S}{\partial p_{i'}} \Big|_{t=t_i^S} & \dots \text{ if } \tilde{\mathbf{t}}_i \neq \emptyset \text{ and } \sum_{i' \in \tilde{\mathbf{t}}_i} \frac{\partial p_i^S}{\partial p_{i'}} \neq 0 \\ \sum_{j \in \tilde{\mathbf{t}}_i^{\subset S}} \frac{\partial p_i^S}{\partial p_j} \Big|_{t=t_i^S} \bullet f_j(t_i^S | \tilde{\boldsymbol{\theta}}_j) & \dots \text{ otherwise.} \end{array} \right] \quad \text{---(148)}
\end{aligned}$$

given $\tilde{\mathbf{t}}_i = \{\forall i'\}$ such that $t_{i'}^S = t_i^S$ and $i' \in \tilde{\mathbf{t}}_i^{\subset S}$, where p_i^S is the probability of the i^{th} sensor detecting failure expressed as a function of time t , *inferentially subordinate* component and sensor failure (detection) probabilities p_j and $p_{i'}^S$,

$$\begin{aligned}
p_{i'}^S \Big|_t &= \left\{ \begin{array}{ll} \frac{f_{i'}^S(t | \tilde{\mathbf{t}}_i^{\subset S}, \boldsymbol{\theta}_i^{\subset S})}{1 - F_{i'}^S(t_i^* | \tilde{\mathbf{t}}_i^{\subset S}, \boldsymbol{\theta}_i^{\subset S})} & \dots \text{ if } t_i^S = \{\forall t > t_i^*\} \text{ and } t > t_i^* \\ H(t - t_{i'}^S) & \dots \text{ otherwise} \end{array} \right\}, \quad p_j \Big|_t = F_j(t | \tilde{\boldsymbol{\theta}}_j) \quad \text{and} \\
\frac{dp_j}{dt} \Big|_t &= f_j(t | \tilde{\boldsymbol{\theta}}_j).
\end{aligned}$$

6.6. CONTINUOUS LIFE METRIC SYSTEM - ALGORITHM

The practical construction of the likelihood function, equation (144), is based on the distinct steps listed below. It is assumed that the logic of the system in question has been determined (including sensor placement), and all components have been modelled on respective parameter sets.

6.6.1. Step 1: Determine the set of all inferentially subordinate components and sensors for each sensor.

This involves all sensors. The sets for the i th sensor are:

$\tilde{\mathbf{i}}_i^{\subset S}$... the set of indices of all sensors that are *inferentially subordinate* to the i^{th} sensor; and
 $\tilde{\mathbf{j}}_i^{\subset S}$... the set of indices of all components that are *inferentially subordinate* to the i^{th} sensor.

For example, if a system fails whenever a sub-system with its own sensor (sensor #2) or a separate component (component 1) fails, then the relevant sets for sensor #1 (the system) are $\tilde{\mathbf{i}}_1^{\subset S} = \{2\}$ and $\tilde{\mathbf{j}}_1^{\subset S} = \{1\}$.

6.6.2. Step 2: Model system failure detection probabilities on inferentially subordinate components and sensors.

This involves generating expressions for each sensor, as per equation (139). The function, by definition, must be expressed in terms of the *inferentially subordinate* sensors and components established in the previous step.

$$F_i^S(t | \boldsymbol{\theta}_i^{\subset S}, \tilde{\mathbf{i}}_i^{\subset S}) = p_i^S = p_i^S(\forall p_i^S, \forall p_j) \quad \text{---(139)}$$

The structure of these functions is based on system logic and derived utilizing Boolean algebra. For example, if a system fails whenever a sub-system with its own sensor (sensor #2) or a

separate component (component 1) fails, then the *probability of failure detection* for the 1st sensor (the system) is defined as $p_1^S = p_2^S + p_1 - (p_2^S p_1)$. All inferentially subordinate sensors that appear on the right hand side of these equations are denoted $p_{i'}^S$ (and therefore $i' \in \tilde{\mathbf{t}}_1^{\subset S}$).

6.6.3. Step 3: Compile evidence.

The evidence set, $\tilde{\mathbf{t}}^S = \{t_1^S, t_2^S, t_3^S, \dots, t_i^S, \dots, t_m^S\}$, is the compilation of times until failure detection for each sensor, t_i^S . If the detection time is right censored (as in detection stopped at t_i^* with no failure), then t_i^S is the set of all time greater than t_i^* : $t_i^S = \{\forall t > t_i^*\}$.

6.6.4. Step 4: Compile evidential probability of failure detection $p_{i'}^S$ for all inferentially subordinate sensors.

The expression for $p_{i'}^S$ is conditional on the evidence set.

$$p_{i'}^S \Big|_t = \begin{cases} \frac{f_{i'}^S(t | \tilde{\mathbf{t}}_{i'}^{\subset S}, \boldsymbol{\theta}_{i'}^{\subset S})}{1 - F_{i'}^S(t_i^* | \tilde{\mathbf{t}}_{i'}^{\subset S}, \boldsymbol{\theta}_{i'}^{\subset S})} & \dots \quad \text{if } t_{i'}^S = \{\forall t > t_i^*\} \text{ and } t > t_i^* \\ H(t - t_i^S) & \dots \quad \text{otherwise} \end{cases} \quad \text{---(147)}$$

6.6.5. Step 5: Differentiate all sensor failure detection probabilities, p_i^S , with respect to inferentially subordinate sensor failure detection probabilities, $p_{i'}^S$.

For example, if a system fails whenever a sub-system with its own sensor (sensor #2) or a separate component (component 1) fails, then the derivative of p_1^S with respect to p_2^S is $1 - p_1$.

6.6.6. Step 6: Differentiate all sensor failure detection probabilities, p_i^S , with respect to *inferentially subordinate* component failure probabilities, p_j .

For example, if a system fails whenever a sub-system with its own sensor (sensor #2) or a separate component (component 1) fails, then the derivative of p_1^S with respect to p_1 is $1 - p_2^S$.

6.6.7. Step 7: For all sensors, compile set of all *inferentially subordinate* sensors that have identical failure detection times.

That is, $\tilde{i}_i = \{\forall i'\}$ such that $t_{i'}^S = t_i^S$ and $i' \in \tilde{i}_i^{CS}$. It is possible that no *inferentially subordinate* sensors share the same time to failure detection to the i^{th} sensor, in which case \tilde{i}_i will be a null set, \emptyset .

6.6.8. Step 8: Substitute all elements into the likelihood function.

At this stage, all elements of equation (148) have been determined, and substitution will yield the likelihood function.

Example 18: Basic two component continuous life metric system *overlapping data* analysis

Consider the simple parallel system illustrated in Figure 44. The system is tested until failure is detected by each sensor. Sensor #2 detects failure first, followed by the system (sensor #1). Both components have had their time to failure probability modelled and is summarized in Table 14.

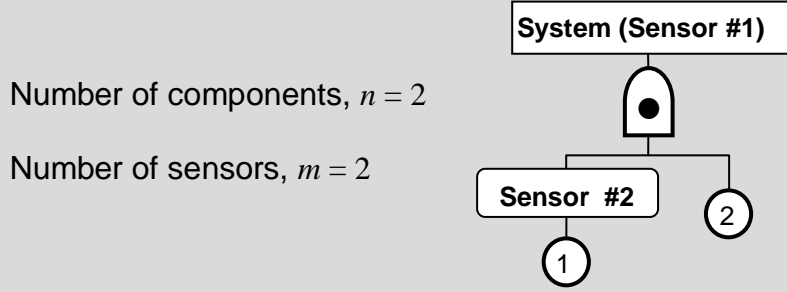


Figure 44: Basic two component parallel system

| Component number: j | PDF: $f_j(t \tilde{\theta}_j)$ | CDF: $F_j(t \tilde{\theta}_j)$ | Parameters: $\tilde{\theta}_j$ |
|-----------------------|--------------------------------|--------------------------------|--------------------------------|
| 1 | $\lambda_1 e^{-\lambda_1 t}$ | $1 - e^{-\lambda_1 t}$ | λ_1 |
| 2 | $\lambda_2 e^{-\lambda_2 t}$ | $1 - e^{-\lambda_2 t}$ | λ_2 |

Table 14: Component reliability characteristics for system illustrated in **Figure 44**.

Step 1. The sets of *inferentially subordinate* sensors and components for each sensor are shown in Table 15:

| Sensor #: i | $\tilde{\mathbf{i}}_i^{\subset S}$ | $\tilde{\mathbf{j}}_i^{\subset S}$ |
|---------------|------------------------------------|------------------------------------|
| 1 | $\{2\}$ | $\{2\}$ |
| 2 | \emptyset | $\{1\}$ |

Table 15: Sets of *inferentially subordinate* sensors and components for the system in **Figure 44**.

Step 2. The sensor failure detection probabilities in terms of *inferentially subordinate* components and sensors are listed in equations (149) and (150).

$$\text{For sensor \#1: } p_1^S = p_2^S \times p_2 \quad \text{---(149)}$$

$$\text{For sensor \#2: } p_2^S = p_1 \quad \text{---(150)}$$

Step 3. The times to failure detection generate the evidence set:

$$E = \tilde{t}^S = \{t_1^S = 10.538 \text{ h}, t_2^S = 9.269 \text{ h}\} \quad \text{---(151)}$$

Step 4. There is only one *inferentially subordinate* sensor (sensor #2), and its evidential failure probability is:

$$p_2^S \Big|_t = H(t - 9.269) \quad \text{---(152)}$$

Steps 5 and 6. All relevant partial derivatives of higher level probabilities are shown in Table 16:

| x | $\frac{\partial p_1^S}{\partial x}$ | $\frac{\partial p_2^S}{\partial x}$ |
|---------|-------------------------------------|-------------------------------------|
| p_2^S | p_2 | n/a |
| p_1 | n/a | 1 |
| p_2 | p_2^S | n/a |

Table 16: Partial derivatives of higher level failure probabilities for the missile guidance system

Step 7. There are no instances of identical times to failure detection, so $\tilde{t}_i = \emptyset$ for all i .

Step 8. Substituting the above equations into equation (148) yields:

$$\begin{aligned}
 L(\tilde{\mathbf{t}}^S | \boldsymbol{\theta}) &\equiv \prod_{i=1}^2 \left[\begin{array}{ll} 1 - p_i^S \Big|_{t=t_i^*} & \dots \text{ if } t_i^S = \{\forall t > t_i^*\} \text{ (i.e. right censored data)} \\ \sum_{i' \in \tilde{\mathbf{i}}_i} \frac{\partial p_i^S}{\partial p_{i'}} \Big|_{t=t_i^S} & \dots \text{ if } \tilde{\mathbf{i}}_i \neq \emptyset \text{ and } \sum_{i' \in \tilde{\mathbf{i}}_i} \frac{\partial p_i^S}{\partial p_{i'}} \neq 0 \\ \sum_{j \in \tilde{\mathbf{j}}_i^{cS}} \frac{\partial p_i^S}{\partial p_j} \Big|_{t=t_i^S} \bullet f_j(t_i^S | \tilde{\boldsymbol{\theta}}_j) & \dots \text{ otherwise.} \end{array} \right] \\
 &= \prod_{i=1}^2 \left(\sum_{j \in \tilde{\mathbf{j}}_i^{cS}} \frac{\partial p_i^S}{\partial p_j} \Big|_{t=t_i^S} \bullet f_j(t_i^S | \tilde{\boldsymbol{\theta}}_j) \right) \begin{array}{l} \text{as there are no instances of right} \\ \text{censoring, and no instances of identical} \\ \text{times to failure detection at multiple} \\ \text{sensors.} \end{array} \\
 &= \left(\sum_{j \in \tilde{\mathbf{j}}_1^{cS}} \frac{\partial p_1^S}{\partial p_j} \Big|_{t=t_1^S} \bullet f_j(t_1^S | \tilde{\boldsymbol{\theta}}_j) \right) \left(\sum_{j \in \tilde{\mathbf{j}}_2^{cS}} \frac{\partial p_2^S}{\partial p_j} \Big|_{t=t_2^S} \bullet f_j(t_2^S | \tilde{\boldsymbol{\theta}}_j) \right) \\
 &= \left(\frac{\partial p_1^S}{\partial p_2} \Big|_{t=t_1^S} \bullet f_2(t_1^S | \tilde{\boldsymbol{\theta}}_2) \right) \left(\frac{\partial p_2^S}{\partial p_1} \Big|_{t=t_2^S} \bullet f_1(t_2^S | \tilde{\boldsymbol{\theta}}_1) \right) \\
 &= \left(p_2^S \Big|_{t=t_1^S} \bullet f_2(t_1^S | \tilde{\boldsymbol{\theta}}_2) \right) \left(f_1(t_2^S | \tilde{\boldsymbol{\theta}}_1) \right) \\
 &= \left(\lambda_2 e^{-\lambda_2 t_1^S} \right) \left(\lambda_1 e^{-\lambda_1 t_2^S} \right) = \lambda_1 \lambda_2 e^{-(9.269\lambda_1 + 10.538\lambda_2)} \quad \text{---(153)}
 \end{aligned}$$

It can be observed that the likelihood function in equation (153) is equivalent to:

$$L(\tilde{\mathbf{t}}^S | \boldsymbol{\theta}) = \Pr \left\{ \begin{array}{l} \text{component 1 failing at 9.269 h given } \lambda_1; \text{ and} \\ \text{component 2 failing at 10.538 h given } \lambda_2 \end{array} \right. \quad \text{---(154)}$$

The likelihood function is illustrated in Figure 45.

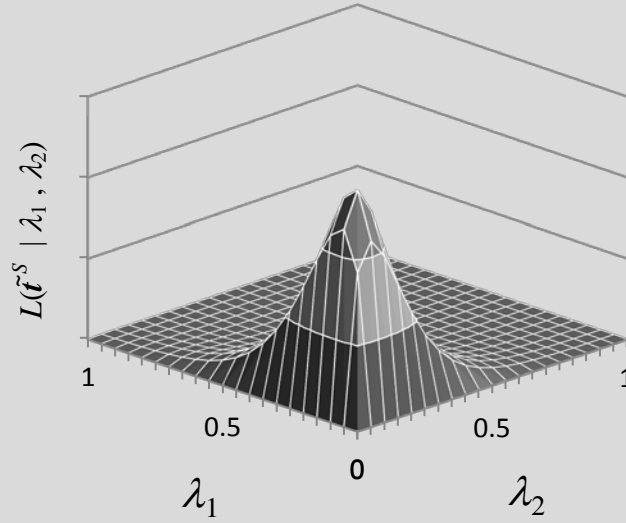


Figure 45: Likelihood function for system illustrated in **Figure 44** with evidence set (151)

The *downwards inference* technique outlined above is able to incorporate *overlapping data* from various levels from within a system and can identify situations where sensor data is able to infer information directly about individual component failure times (a situation explored in Example 18). *Downwards inference* also infers information when the sequence of sensor failure detection effectively ‘masks’ some of the component failure times (a situation explored in Example 19).

Example 19: Basic two component continuous life metric system *overlapping data* analysis

Consider the same simple parallel system considered in Example 18 and illustrated in Figure 44. The system is tested until failure is detected by each sensor. As opposed to Example 18, the system (sensor #1) and sensor #2 detect failures simultaneously. The failure detection times, in equation (155), generate the evidence set.

$$E = \tilde{t}^S = \{t_1^S = t_2^S = 9.745 \text{ h}\} \quad \text{---(155)}$$

Steps 1, 2, 5 and 6 in developing the likelihood function are identical to those in Example 18. Modified steps are listed below.

Step 3: The times to failure detection generate the evidence set are $E = \tilde{t}^S = \{t_1^S = t_2^S = 9.745 \text{ h}\}$.

Step 4. There is only one *inferentially subordinate* sensor (sensor #2), and its evidential failure probability is:

$$p_{2'}^S \Big|_t = H(t - 9.745) \quad \text{---(156)}$$

Step 7. Since $t_1^S = t_2^S$: $\tilde{i}_1 = \{2\}$ ---(157)

But $\tilde{i}_2 = \emptyset$ as sensor #2 has no *inferentially subordinate* sensors ---(158)

Step 8. Substituting the above equations into equation (148) yields equation (159).

$$\begin{aligned}
L(\tilde{\mathbf{t}}^S | \boldsymbol{\theta}) &\equiv \prod_{i=1}^2 \left[\begin{array}{ll} 1 - p_i^S \Big|_{t=t_i^*} & \dots \text{ if } t_i^S = \{\forall t > t_i^*\} \text{ (i.e. right censored data)} \\ \sum_{i' \in \tilde{\mathbf{t}}_i} \frac{\partial p_i^S}{\partial p_{i'}} \Big|_{t=t_i^S} & \dots \text{ if } \tilde{\mathbf{t}}_i \neq \emptyset \text{ and } \sum_{i' \in \tilde{\mathbf{t}}_i} \frac{\partial p_i^S}{\partial p_{i'}} \neq 0 \\ \sum_{j \in \tilde{\mathbf{j}}_i^S} \frac{\partial p_i^S}{\partial p_j} \Big|_{t=t_i^S} \bullet f_j(t_i^S | \tilde{\boldsymbol{\theta}}_j) & \dots \text{ otherwise.} \end{array} \right] \\
&= \left(\sum_{i' \in \tilde{\mathbf{t}}_1} \frac{\partial p_1^S}{\partial p_{i'}} \Big|_{t=t_1^S} \right) \left(\sum_{j \in \tilde{\mathbf{j}}_2^S} \frac{\partial p_2^S}{\partial p_j} \Big|_{t=t_2^S} \bullet f_j(t_2^S | \tilde{\boldsymbol{\theta}}_j) \right) \\
&= \left(\frac{\partial p_1^S}{\partial p_2^S} \Big|_{t=t_1^S} \right) \left(\frac{\partial p_2^S}{\partial p_1} \Big|_{t=t_1^S} \bullet f_1(t_2^S | \tilde{\boldsymbol{\theta}}_1) \right) = (p_2|_{t=t_1^S}) (1 \bullet f_1(t_2^S | \tilde{\boldsymbol{\theta}}_1)) \\
&= F_2(t_1^S | \tilde{\boldsymbol{\theta}}_2) f_1(t_2^S | \tilde{\boldsymbol{\theta}}_1) = (1 - e^{-\lambda_2 t_1^S}) \lambda_1 e^{-\lambda_1 t_2^S} \\
&= (1 - e^{-9.745 \lambda_2}) \lambda_1 e^{-9.745 \lambda_1} \quad \text{---(159)}
\end{aligned}$$

It can be observed that the likelihood function in equation (159) is equivalent to:

$$L(\tilde{\mathbf{t}}^S | \boldsymbol{\theta}) = \Pr \left\{ \begin{array}{l} \text{component 1 failing at 9.745 h given } \lambda_1; \text{ and} \\ \text{component 2 failing before 9.745 h given } \lambda_2 \end{array} \right. \quad \text{---(160)}$$

The likelihood function is illustrated in Figure 46.

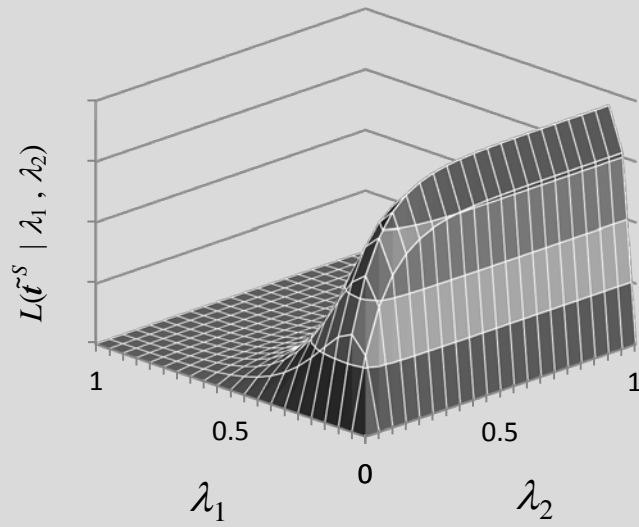
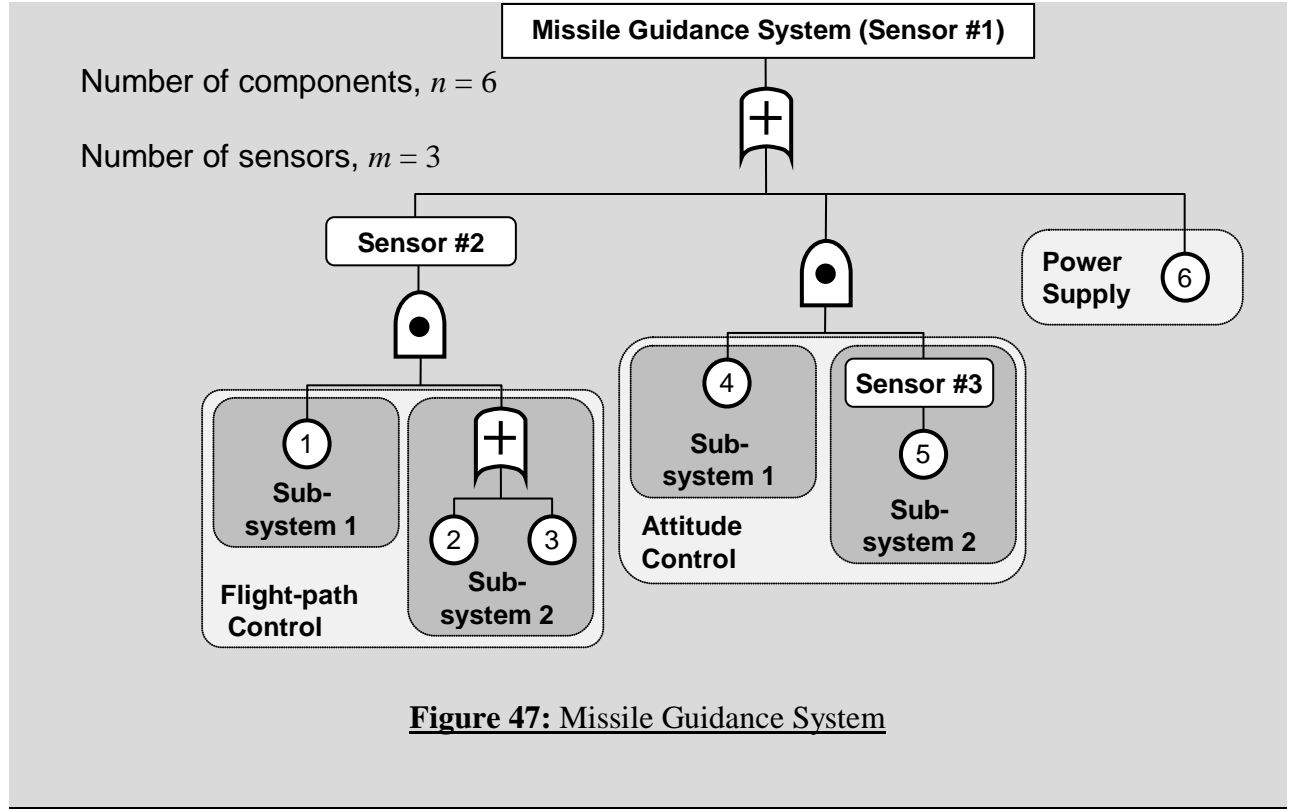


Figure 46: Likelihood function for system illustrated in **Figure 44** with evidence set (155)

Example 20: Basic two component continuous life metric system *overlapping data* analysis

Consider the simplified missile guidance system illustrated in Figure 47. It is being tested on inert missiles that have three additional sensors embedded that relay information back to a ground station real-time. The six components have had their time to failure probability modelled, and prior information exists for some of their reliability parameters from previous testing regimes and expert solicitation. This modelling and information is summarized in Table 17.



| Component number j | PDF $f_j(t \tilde{\theta}_j)$ | CDF $F_j(t \tilde{\theta}_j)$ | Parameters $\tilde{\theta}_j$ | Prior Distribution |
|----------------------|--|--|----------------------------------|---|
| 1* | $\lambda_1 e^{-\lambda_1 t}$ | $1 - e^{-\lambda_1 t}$ | λ_1 | $\pi_0(\lambda_1) = \begin{cases} 1/\sqrt{\lambda_1} & \dots & \lambda_1 > 0 \\ 0 & \dots & \lambda_1 \leq 0 \end{cases}$ |
| 2 | $\frac{\beta t^{\beta-1}}{\eta^\beta} e^{-\left(\frac{t}{\eta}\right)^\beta}$ | $1 - e^{-\left(\frac{t}{\eta}\right)^\beta}$ | β_2 | $\pi_0(\beta_2) = U(2,3)$ |
| | | | η_2 | $\pi_0(\eta_2) = U(100,150)$ |
| 3* | $\frac{e^{-\frac{1}{2}\left(\frac{t-\frac{3}{\sigma_3}}{\sigma_3}\right)^2}}{\sqrt{2\pi\sigma_3^2}}$ | $\frac{1}{2} \left[1 + \operatorname{erf} \left(\frac{t-\frac{3}{\sigma_3}}{\sigma_3\sqrt{2}} \right) \right]$ | μ_3 | $\pi_0(\mu_3) = \begin{cases} 1 & \dots & \mu_3 > 0 \\ 0 & \dots & \mu_3 \leq 0 \end{cases}$ |
| | | | σ_3 | $\pi_0(\sigma_3) = \begin{cases} 1/\sigma_3 & \dots & \sigma_3 > 0 \\ 0 & \dots & \sigma_3 \leq 0 \end{cases}$ |
| 4 | $\lambda_4 e^{-\lambda_4 t}$ | $1 - e^{-\lambda_4 t}$ | λ_4 | $\pi_0(\lambda_4) = U(0.01,0.03)$ |
| 5 | $\lambda_5 e^{-\lambda_5 t}$ | $1 - e^{-\lambda_5 t}$ | λ_5 | $\pi_0(\lambda_5) = U(0,0.02)$ |
| 6 | $\lambda_6 e^{-\lambda_6 t}$ | $1 - e^{-\lambda_6 t}$ | λ_6 | $\pi_0(\lambda_6) = U(0,0.1)$ |

Table 17: Missile Guidance System component reliability characteristics for time expressed as minutes (* - Jeffreys' non-informative prior distributions for relevant parameters [19]).

The entire set of parameters is defined as: $\theta = \{\lambda_1, \beta_2, \eta_2, \mu_3, \sigma_3, \lambda_4, \lambda_5, \lambda_6\}$ ---(161)

Step 1. The set of *inferentially subordinate* sensors and components for each sensor is shown in Table 18.

| Sensor # i | \tilde{i}_i^S | \tilde{j}_i^S |
|--------------|-----------------|-----------------|
| 1 | {2,3} | {4,6} |
| 2 | \emptyset | {1,2,3} |
| 3 | \emptyset | {5} |

Table 18: Sets of *inferentially subordinate* sensors and components for the system in Figure 47

Step 2. The sensor failure detection probabilities in terms of *inferentially subordinate* components and sensors are:

$$\text{For sensor \#1: } p_1^S = p_2^S + p_4 p_3^S + p_6 - p_2^S p_4 p_3^S - p_2^S p_6 - p_4 p_3^S p_6 + p_2^S p_4 p_3^S p_6 \quad \text{---(162)}$$

$$\text{For sensor \#2: } p_2^S = p_1 (p_2 + p_3 - p_2 p_3) \quad \text{---(163)}$$

$$\text{For sensor \#3: } p_3^S = p_5 \quad \text{---(164)}$$

Step 3. The missile guidance system was tested until it failed at 113.54 minutes. Before failure, sensor #3 detected failure at 78.69 minutes. After the system failed, the inert missile lost control causing the test to be declared complete at the time of system failure, with sensor #2 yet to detect failure:

$$E = \tilde{t}^S = \{t_1^S = 113.54 \text{ min}, t_2^S > 113.54 \text{ min}, t_3^S = 78.69 \text{ min}\} \quad \text{---(165)}$$

Step 4. The evidential failure probabilities of the *inferentially subordinate* sensors (sensor #2 and sensor #3) are listed in equation (166).

$$p_{2'}^S \Big|_t = \begin{cases} \frac{f_2^S(t | \tilde{t}_{2'}^{cS}, \theta_{2'}^{cS})}{1 - F_2^S(t_i^* | \tilde{t}_{2'}^{cS}, \theta_{2'}^{cS})} & \dots \text{ if } t > 113.54 \\ 0 & \dots \text{ otherwise} \end{cases} \quad \text{---(166)}$$

noting that as the test concludes at 113.54 minutes, there is no need to develop equation (166) further.

$$p_{3'}^S \Big|_t = H(t - 78.69) \quad \text{---(167)}$$

Steps 5 and 6. All relevant partial derivatives of higher level probabilities are shown in can be calculated:

| X | $\frac{\partial p_1^S}{\partial x}$ | $\frac{\partial p_2^S}{\partial x}$ | $\frac{\partial p_3^S}{\partial x}$ |
|------------|---|-------------------------------------|-------------------------------------|
| $p_{2'}^S$ | $1 + p_4 p_{3'}^S - p_6 + p_4 p_{3'}^S p_6$ | n/a | n/a |
| $p_{3'}^S$ | $p_4 - p_{2'}^S p_4 - p_4 p_6 + p_{2'}^S p_4 p_6$ | n/a | n/a |
| p_1 | n/a | $p_2 + p_3 - p_2 p_3$ | n/a |
| p_2 | n/a | $p_1(1 - p_3)$ | n/a |
| p_3 | n/a | $p_1(1 - p_2)$ | n/a |
| p_4 | $p_{3'}^S - p_{2'}^S p_{3'}^S - p_{3'}^S p_6 + p_{2'}^S p_{3'}^S p_6$ | n/a | n/a |
| p_5 | n/a | n/a | 1 |
| p_6 | $1 - p_{2'}^S - p_4 p_{3'}^S + p_{2'}^S p_4 p_{3'}^S$ | n/a | n/a |

Table 19: Partial derivatives of higher level failure probabilities for the missile guidance system

Step 7. There are no instances of identical times to failure detection, so $\mathbf{i}_i = \emptyset$ for all i .

Step 8. Substituting the above equations into equation (148) yields:

$$\begin{aligned}
 L(\tilde{\mathbf{t}}^S | \boldsymbol{\theta}) &\equiv \prod_{i=1}^3 \left[\begin{array}{ll} 1 - p_i^S \Big|_{t=t_i^*} & \dots \text{ if } t_i^S = \{\forall t > t_i^*\} \text{ (i.e. right censored data)} \\ \sum_{i' \in \tilde{\mathbf{i}}_i} \frac{\partial p_i^S}{\partial p_{i'}} \Big|_{t=t_i^S} & \dots \text{ if } \tilde{\mathbf{i}}_i \neq \emptyset \text{ and } \sum_{i' \in \tilde{\mathbf{i}}_i} \frac{\partial p_i^S}{\partial p_{i'}} \neq 0 \\ \sum_{j \in \tilde{\mathbf{j}}_i^S} \frac{\partial p_i^S}{\partial p_j} \Big|_{t=t_i^S} \bullet f_j(t_i^S | \tilde{\boldsymbol{\theta}}_j) & \dots \text{ otherwise.} \end{array} \right] \\
 &\equiv \left\{ \begin{array}{l} \left(\sum_{j \in \tilde{\mathbf{j}}_1^S} \frac{\partial p_1^S}{\partial p_j} \Big|_{t=113.54} \bullet f_j(113.54 | \tilde{\boldsymbol{\theta}}_j) \right) \\ \times \left(1 - p_2^S \Big|_{t=113.54} \right) \\ \times \left(\sum_{j \in \tilde{\mathbf{j}}_3^S} \frac{\partial p_3^S}{\partial p_j} \Big|_{t=78.69} \bullet f_j(78.69 | \tilde{\boldsymbol{\theta}}_j) \right) \end{array} \right\} \\
 &\equiv \left\{ \begin{array}{l} \left(\left(\left(p_3^S - p_2^S p_3^S - p_3^S p_6 + p_2^S p_3^S p_6 \Big|_{t=113.54} \right) \bullet f_4(113.54 | \tilde{\boldsymbol{\theta}}_4) \right) \right) \\ + \left(\left(1 - p_2^S - p_4 p_3^S + p_2^S p_4 p_3^S \Big|_{t=113.54} \right) \bullet f_6(113.54 | \tilde{\boldsymbol{\theta}}_6) \right) \\ \times \left(1 - p_1(p_2 + p_3 - p_2 p_3) \Big|_{t=113.54} \right) \bullet f_5(78.69 | \tilde{\boldsymbol{\theta}}_5) \end{array} \right\} \quad \text{---(168)}
 \end{aligned}$$

Substituting equations (166) and (167) into equation (168) yields equation (169).

$$L(\tilde{\mathbf{t}}^S | \boldsymbol{\theta}) \equiv \left\{ \begin{array}{l} \left(\left((1 - p_6|_{t=113.54}) \bullet f_4(113.54 | \tilde{\boldsymbol{\theta}}_4) \right) \right. \\ \left. + \left((1 - p_4|_{t=113.54}) \bullet f_6(113.54 | \tilde{\boldsymbol{\theta}}_6) \right) \right) \\ \times \left(1 - p_1(p_2 + p_3 - p_2 p_3)|_{t=113.54} \right) \bullet f_5(78.69 | \tilde{\boldsymbol{\theta}}_5) \end{array} \right\} \quad \text{---(169)}$$

Breaking down equation (169) illustrates the nature of the system failure. In essence, the likelihood function in equation (169) is equivalent to:

$$L(\tilde{\mathbf{t}}^S | \boldsymbol{\theta}) = \Pr \left\{ \begin{array}{l} \left(\begin{array}{l} \text{components 6 functional and 4 failing at 113.54 given } \{\lambda_4, \lambda_6\}; \text{ or} \\ \text{components 4 functional and 6 failing at 113.54 given } \{\lambda_4, \lambda_6\} \end{array} \right); \text{ and} \\ \text{sensor \#2 not detecting failure at 113.54 given } \{\lambda_1, \beta_2, \eta_2, \mu_3, \sigma_3\}; \text{ and} \\ \text{component 5 failing at 78.69 given } \lambda_5 \end{array} \right\}$$

Substitution into Bayes' Theorem yields:

$$\pi_1(\boldsymbol{\theta} | \tilde{\mathbf{t}}^S) = \frac{L(\tilde{\mathbf{t}}^S | \boldsymbol{\theta}) \pi_0(\boldsymbol{\theta})}{\int_{\forall \boldsymbol{\theta}^6} L(\tilde{\mathbf{t}}^S | \boldsymbol{\theta}') \pi_0(\boldsymbol{\theta}') d\boldsymbol{\theta}'} \propto L(\tilde{\mathbf{t}}^S | \boldsymbol{\theta}) \pi_0(\boldsymbol{\theta}) \quad \text{---(170)}$$

where $L(\tilde{\mathbf{t}}^S | \boldsymbol{\theta})$ is provided by equation (169) and $\pi_0(\boldsymbol{\theta})$ is given by the product of all prior distributions defined in Table 17.

A set of random draws utilizing Markov chain Monte Carlo simulation is an effective way of understanding the nature of the joint posterior distributions of the reliability parameters. [20] The following statistics are drawn for each parameter:

| Parameters θ | Percentiles | | | | | |
|------------------------|-----------------------------------|------------------|------------------|-----------------|------------------|------------------|
| | Prior | | | Posterior | | |
| | 5 th | 50 th | 95 th | 5 th | 50 th | 95 th |
| λ_1 | n/a – improper prior distribution | | | 0.0396 | 3.0383 | 17.375 |
| β_2 | 2.05 | 2.50 | 2.95 | 2.0757 | 2.5019 | 2.9270 |
| η_2 | 102.5 | 125.0 | 147.5 | 104.98 | 128.72 | 147.22 |
| μ_3 | n/a – improper prior distribution | | | 12.511 | 137.49 | 240.25 |
| σ_3 | n/a – improper prior distribution | | | 45.464 | 135.02 | 210.13 |
| λ_4 | 0.011 | 0.020 | 0.029 | 0.0108 | 0.0168 | 0.0272 |
| λ_5 | 0.001 | 0.010 | 0.019 | 0.0027 | 0.0110 | 0.0185 |
| λ_6 | 0.005 | 0.050 | 0.095 | 0.0018 | 0.0132 | 0.0446 |

Table 20: Missile Guidance System reliability parameter statistics (derived from Markov chain Monte Carlo simulation with 100 000 draws).

6.7. SUMMARY

Fully Bayesian methodologies have been developed for *overlapping data* at various levels within on-demand systems. The basis of this methodology is referred to as *downwards inference* and is extended to systems based on continuous life metrics in this chapter. A key aspect of *downwards inference* is the ability to incorporate *overlapping data*. Constraining *overlapping data* as non-overlapping ignores the dependencies between the data sets and effectively removes information. An *overlapping data* likelihood function was developed to incorporate these inherent dependencies and generate the correct inference within Bayes' Theorem for systems. All examples and equations were time based, but can easily transferrable to any other continuous independent random variable such as distance. The methodology developed above allow all information gathered from various hierarchical levels within a system to be correctly analysed to infer all facets of information that such *overlapping data* sets contain. Several examples were developed to highlight the effect of the additional information *overlapping data* contains and

how it can be used to correctly improve our state of knowledge (which is the set of component reliability characteristics parameters). The flexibility of the likelihood function allows incorporation of multiple instances of the same component simultaneously. Through state of knowledge dependence, the resultant *overlapping data* Bayesian method completely incorporates all information and evidence that can possible be generated or observed by complex time based systems.

Chapter 7: Uncertain Evidence

7.1. INTRODUCTION

This chapter explores fully Bayesian methodologies for incorporating uncertain *overlapping data* based on the likelihood functions outlined in previous chapters. In the case of on-demand systems, uncertain data manifests itself in terms of the number of observed degraded states from a number of demands, whereas it is manifested in terms of the time at which failure is detected for continuous time based systems. Incorporating uncertain evidence in the latter yields a likelihood function that is not only computationally simpler than that proposed in chapter 6, but correctly replicates reality in that all time detection devices have a known uncertainty.

7.2. GENERAL FRAMEWORK FOR UNCERTAIN DATA

Uncertain data in a Bayesian context conventionally refers to error associated with data collection. A system will behave in a particular way, and uncertainty is drawn from any process that prevents this particular way to be properly understood. The observed (uncertain) evidence, \hat{E} , is separate to the ‘true’ (unknown) evidence, E . When the observed evidence, \hat{E} , is gathered using a process or device that has an inherent deviation from the actual evidence, E , the relationship between the two (or uncertainty) is probabilistic. Bayesian analysis necessarily requires this uncertainty to be characterized by a subjective probabilistic relationship.

There are two types of methods that exist when the uncertainty is expressed as a conditional probability. The first type is based on the conditional probability that E is the true evidence set

when the evidence set \hat{E} is observed: $\Pr(E | \hat{E})$. This probabilistic relationship is referred to as *Berkson error* [21-23]. The first method for dealing with evidence uncertainty is Jeffrey's rule of probability kinematics [24], and revolves around the generation of a posterior distribution which itself is the weighted average or weighted sum of posterior distributions for all possible true evidence sets, E as shown in equation (171).

$$\pi_1(\boldsymbol{\theta} | \hat{E}) = \sum_{\forall E} \pi_1(\boldsymbol{\theta} | E) \Pr(E | \hat{E}) \quad \text{---(171)}$$

Cheeseman's rule [25] is based on theory where prior information and observed data is combined to construct a weighted likelihood function, and is shown in equation (172).

$$L(\hat{E} | \boldsymbol{\theta}) = \sum_{\forall E} L(E | \boldsymbol{\theta}) \Pr(E | \hat{E}) \quad \text{---(172)}$$

Tan and Xi [26] propose a method where *Classical error* is used to summarize the relationship between E and \hat{E} . Classical error expresses the uncertainty as a conditional probability that \hat{E} is the observed evidence set when the true evidence set is E : $\Pr(\hat{E} | E)$. [21, 23, 27] The method, referred to as *likelihood in terms of observation*, is expressed in equation (173).

$$\text{---(173)}$$

As mentioned in previous chapters, the calculation of the likelihood function of on-demand systems using downwards inference is computationally intensive. Therefore, there will be a significant computational liability with treating uncertainty evidence through evaluations of equation (171), (172) or (173).

7.3. EVIDENCE UNCERTAINTY FOR ON-DEMAND SYSTEMS.

With the nomenclature outlined above for both observed and true evidence sets, the likelihood function for *overlapping data sets* from multi-state on-demand systems in equation (64) can be re-written as equation (174).

$$L(\hat{E} | \mathbf{p}) \propto \sum_{\forall \tilde{\mathbf{v}}_a \in \mathbf{v}_{\hat{E}}} \left[\prod_{l=1}^{z^{n'}} \left\{ \frac{1}{(v_l)_a!} \left(\prod_{b=1}^{n'} p_{j_b}^{((x_b)_l)} \right)^{(v_l)_a} \right\} \right] \quad \text{---(174)}$$

recalling that when $\tilde{\mathbf{v}}_a$ implies the data/evidence set \hat{E} , it appears in the set \mathbf{v}_E .

In essence the right hand side of the proportionality in equation (174) is simply a sum of the probability of occurrence for all possible combinations of state vectors that imply the observed evidence for r demands. The term $\tilde{\mathbf{v}}_a$ refers to the a^{th} combination of component state vectors, which is equivalent to a possible true evidence set, E_a . Therefore, (174) is expressed in terms of classical error as shown in equation (175).

$$\begin{aligned}
L(\hat{E} | \mathbf{p}) &= \sum_{\forall \tilde{\mathbf{v}}_a \in \mathbf{v}_{\hat{E}}} \Pr(\tilde{\mathbf{v}}_a | \mathbf{p}) = \sum_{\forall \tilde{\mathbf{v}}_a} \Pr(\hat{E} | \tilde{\mathbf{v}}_a) \Pr(\tilde{\mathbf{v}}_a | \mathbf{p}) \\
\text{where } \Pr(\hat{E} | \tilde{\mathbf{v}}_a) &= \begin{cases} 1 & \dots & \tilde{\mathbf{v}}_a \text{ implies } \hat{E} & \text{(i.e. } \tilde{\mathbf{v}}_a \in \mathbf{v}_E) \\ 0 & \dots & \tilde{\mathbf{v}}_a \text{ doesn't imply } \hat{E} & \text{(i.e. } \tilde{\mathbf{v}}_a \notin \mathbf{v}_E) \end{cases} \\
&\equiv \sum_{\forall E_a \rightarrow \hat{E}} \Pr(E_a | \mathbf{p}) = \sum_{\forall E_a} \Pr(\hat{E} | E_a) \Pr(E_a | \mathbf{p}) \\
\text{where } \Pr(\hat{E} | E_a) &= \begin{cases} 1 & \dots & E_a \text{ implies } \hat{E} & \text{(i.e. } E_a \equiv \tilde{\mathbf{v}}_a \in \mathbf{v}_E) \\ 0 & \dots & E_a \text{ doesn't imply } \hat{E} & \text{(i.e. } E_a \equiv \tilde{\mathbf{v}}_a \notin \mathbf{v}_E) \end{cases} \quad \text{---(175)}
\end{aligned}$$

The likelihood function in (174) therefore adheres to the *likelihood in terms of observation* method expressed in (173).

7.4. CONTINUOUS TIME BASED SYSTEMS

Example 19 in chapter 6 highlights the general implication of multiple sensors detecting failure at identical times for complex systems: there is typically a component that has failed prior to that time allowing the failure of a single component to cause multiple simultaneous failure detection. In reality, it is impossible to conclude from sensor whether failure detection was simultaneous due to limits in accuracy. A tenet of probability theory is that it is impossible for two random events to occur at exactly the same time. The time of failure detections in Example 19 was 9.745 h, which implies the actual time to failure detection at each sensor is greater than or equal to 9.745 h but less than 9.746 h (assuming accuracy of 0.001 h in measurement).⁶ In this case, it is possible for component 2 to have failed after component 1 even if both sensors detect times to

⁶ It is typical with most timing devices that time measurements increase ‘at’ increments of the smallest unit of measurement and remain unchanged until the next increment occurs.

failure of 9.745 h (e.g. components 1 and 2 could have failed at approximately 9.7453 h and 9.7457 h respectively). Time measurements are recorded in multiples of basic time intervals that represent accuracy.

$$\text{e.g. } \hat{t} = i \bullet \Delta t, \quad i \in 1, 2, 3, \dots \quad \text{---(176)}$$

where \hat{t} represents an uncertain observation of the underlying random variable, t (applicable to all random variables written throughout this dissertation).

What this implies is that the actual time of failure, t , exists within the following domain:

$$\hat{t} < t \leq \hat{t} + \Delta t \quad \text{---(177)}$$

The interval Δt is often used to define the accuracy of the time measurement device (e.g. hundredths or thousandths of a second is a typical short time frame accuracy metric). Assuming that all time measurements from multiple sensors within a system share common accuracy intervals, the likelihood function can be amended to be expressed in terms of component and sub-system *CDFs* only (i.e. not *PDFs* and hence no differentiation is required). The uncertain data likelihood function is simplified and written in equation (178).

$$\begin{aligned} L(\hat{\mathbf{t}} | \boldsymbol{\theta}, \Delta t) &= L(\{\hat{t}_1^S, \dots, \hat{t}_m^S\} | \tilde{\mathbf{t}}_1, \dots, \tilde{\mathbf{t}}_n, \Delta t) \\ &= \prod_{i=1}^m \Pr(\hat{t}_i^S \leq T_i^S < (\hat{t}_i^S + \Delta t) | \tilde{\boldsymbol{\theta}}_i^{\subseteq S}, \hat{\mathbf{t}}_i^{\subseteq S}) \end{aligned}$$

$$\begin{aligned}
&\equiv \prod_{i=1}^m \left[F_i^S \left(\hat{t}_i^S + \Delta t \mid \tilde{\theta}_i^{\subset S}, \hat{\mathbf{t}}_i^{\subset S} \right) - F_i^S \left(\hat{t}_i^S \mid \tilde{\theta}_i^{\subset S}, \hat{\mathbf{t}}_i^{\subset S} \right) \right] \\
&= \prod_{i=1}^m \left[p_i^S \Big|_{t=\hat{t}_i^S + \Delta t} - p_i^S \Big|_{t=\hat{t}_i^S} \right] \quad \text{---(178)}
\end{aligned}$$

where Δt is the time increment used in measuring time to failure detection, $\hat{\mathbf{t}}^S$ (as opposed to $\tilde{\mathbf{t}}^S$) represents the vector of failure detection times with an inherent uncertainty, Δt , and $\hat{\mathbf{t}}_i^{\subset S}$ is the set of all uncertain failure detection times of sensors that are *inferentially subordinate* to the i^{th} sensor (i.e. the failure detection times of all sensors $\in \mathbf{S}_i^{\subset S}$).

This modifies the steps (6.6.1 to 6.6.8) to the following steps.

7.4.1. Step 1: Determine the set of all inferentially subordinate components and sensors for each sensor.

The sets for the i^{th} sensor remain unchanged:

$\tilde{\mathbf{i}}_i^{\subset S}$... the set of indices of all sensors that are *inferentially subordinate* to the i^{th} sensor; and

$\tilde{\mathbf{j}}_i^{\subset S}$... the set of indices of all components that are *inferentially subordinate* to the i^{th} sensor.

7.4.2. Step 2: Model system failure detection probabilities on inferentially subordinate components and sensors.

All functions that describe system failure detection probabilities are expressed in terms of the *inferentially subordinate* sensors and components established in the previous step.

$$F_i^S(t | \theta_i^{\subset S}, \tilde{t}_i^{\subset S}) = p_i^S = p_i^S(\forall p_i^S, \forall p_j) \quad \text{---(139)}$$

7.4.3. Step 3: Compile uncertain evidence.

The evidence set (times at which failure is detected) with inherent uncertainty Δt is now represented as $\hat{t}^S = \{\hat{t}_1^S, \hat{t}_2^S, \hat{t}_3^S, \dots, \hat{t}_i^S, \dots, \hat{t}_m^S\}$. As before, if the detection time is right censored (as in detection stopped at \hat{t}_i^* with no failure), then \hat{t}_i^S is the set of all time greater than \hat{t}_i^* : $t_i^S = \{\forall t > \hat{t}_i^*\}$.

7.4.4. Step 4: Compile evidential probability of failure detection $p_{i'}^S$ for all inferentially subordinate sensors.

The expression for $p_{i'}^S$ is conditional on the evidence set.

$$p_{i'}^S \Big|_t = \begin{cases} 0 & \dots & \text{if } t \leq \hat{t}_{i'}^S \\ 1 & \dots & \text{if } t \geq \hat{t}_{i'}^S + \Delta t \end{cases} \quad \text{---(179)}$$

7.4.5. Step 5: Substitute all elements into the likelihood function.

At this stage, all elements of equation (178) have been determined, and substitution will yield the likelihood function.

Example 21: *Overlapping data* analysis of a continuous time-based system with inherent timing inaccuracies.

Consider the same simple parallel system considered in Example 19 based on the same evidence set, but the accuracy of measurement is 0.001 hr. The first two steps remain unchanged, but recall that:

$$\text{For sensor \#1: } p_1^S = p_2^S \times p_2 \quad \text{---(149)}$$

$$\text{For sensor \#2: } p_2^S = p_1 \quad \text{---(150)}$$

The *CDFs* of time to failure for each component (from Table 14) are:

$$\text{Component \#1: } p_1 = F_1(t | \lambda_1) = 1 - e^{-\lambda_1 t} \quad \text{---(180)}$$

$$\text{and Component \#2: } p_2 = F_2(t | \lambda_2) = 1 - e^{-\lambda_2 t} \quad \text{---(181)}$$

Step 3. The times to failure detection generate the evidence set:

$$\hat{E} = \hat{t}^S = \{\hat{t}_1^S = \hat{t}_2^S = 9.745 \text{ h}\} \quad \dots \text{ where } \Delta t = 0.001 \text{ h} \quad \text{---(182)}$$

Step 4. There is only one *inferentially subordinate* sensor (sensor #2), and its evidential failure probability, from equation (179), is shown in equation (183).

$$p_{2'}^S \Big|_t = \begin{cases} 0 & \dots & \text{if } t \leq \hat{t}_2^S \\ 1 & \dots & \text{if } t \geq \hat{t}_2^S + \Delta t \end{cases} \quad \text{---(183)}$$

Step 5: Substituting the above equations into equation (178) yields:

$$\begin{aligned} L(\hat{\mathbf{t}}^S | \boldsymbol{\theta}, \Delta t) &= L(\{\hat{t}_1^S, \hat{t}_2^S\} | \lambda_1, \lambda_2, \Delta t) \\ &= \prod_{i=1}^m \left[p_i^S \Big|_{t=\hat{t}_i^S + \Delta t} - p_i^S \Big|_{t=\hat{t}_i^S} \right] = \left[p_1^S \Big|_{t=\hat{t}_1^S + \Delta t} - p_1^S \Big|_{t=\hat{t}_1^S} \right] \left[p_2^S \Big|_{t=\hat{t}_2^S + \Delta t} - p_2^S \Big|_{t=\hat{t}_2^S} \right] \\ &= \left[(p_2^S \times p_2)_{t=\hat{t}_1^S + \Delta t} - (p_2^S \times p_2)_{t=\hat{t}_1^S} \right] \left[p_1 \Big|_{t=\hat{t}_2^S + \Delta t} - p_1 \Big|_{t=\hat{t}_2^S} \right] \\ &= \left[\left(\begin{cases} 0 & \dots & \text{if } t \leq \hat{t}_2^S \\ 1 & \dots & \text{if } t \geq \hat{t}_2^S + \Delta t \end{cases} \times (1 - e^{-\lambda_2 t}) \right)_{t=\hat{t}_1^S + \Delta t} \right] \left[\begin{matrix} (1 - e^{-\lambda_1 t})_{t=\hat{t}_2^S + \Delta t} \\ -(1 - e^{-\lambda_1 t})_{t=\hat{t}_2^S} \end{matrix} \right] \\ &= \left[\begin{matrix} \begin{cases} 0 & \dots & \text{if } \hat{t}_1^S + \Delta t \leq \hat{t}_2^S \\ 1 - e^{-\lambda_2(\hat{t}_1^S + \Delta t)} & \dots & \text{if } \hat{t}_1^S \geq \hat{t}_2^S \end{cases} \\ - \begin{cases} 0 & \dots & \text{if } \hat{t}_1^S \leq \hat{t}_2^S \\ 1 - e^{-\lambda_2 \hat{t}_1^S} & \dots & \text{if } \hat{t}_1^S \geq \hat{t}_2^S + \Delta t \end{cases} \end{matrix} \right] \left(e^{-\lambda_1 \hat{t}_2^S} - e^{-\lambda_1(\hat{t}_2^S + \Delta t)} \right) \\ &= \begin{cases} \begin{matrix} 0 & \dots & \text{if } \hat{t}_1^S < \hat{t}_2^S \\ 1 - e^{-\lambda_2(\hat{t}_1^S + \Delta t)} & \dots & \text{if } \hat{t}_1^S = \hat{t}_2^S \\ e^{-\lambda_2 \hat{t}_1^S} (1 - e^{-\lambda_2 \Delta t}) & \dots & \text{if } \hat{t}_1^S > \hat{t}_2^S \end{matrix} \end{cases} e^{-\lambda_1 \hat{t}_2^S} (1 - e^{-\lambda_1 \Delta t}) \quad \text{---(184)} \end{aligned}$$

noting that the last step exploits the fact that \hat{t} exists as multiples of Δt .

An observation is that the likelihood function is zero if $\hat{t}_1^S < \hat{t}_2^S$, as it is physically impossible for sensor #1 to detect failure prior to sensor #2 due to it being a series system. Substituting the evidence from (182) generates equation (185).

$$\begin{aligned} L(\hat{t}^S | \theta, \Delta t) &= \left(1 - e^{-\lambda_2(9.745+0.001)}\right) e^{-9.745\lambda_1} \left(1 - e^{-0.001\lambda_1}\right) \\ &= \left(1 - e^{-9.746\lambda_2}\right) e^{-9.745\lambda_1} \left(1 - e^{-0.0001\lambda_1}\right) \end{aligned} \quad \text{---(185)}$$

The same technique was used for the data proposed in Example 18, with the same uncertainty in measurement. The likelihood functions are illustrated in Figure 48. It can be observed that they very closely match the ‘certain’ likelihood functions in Figure 45 and Figure 46, whilst being significantly easier to evaluate (without the need for differentiating).

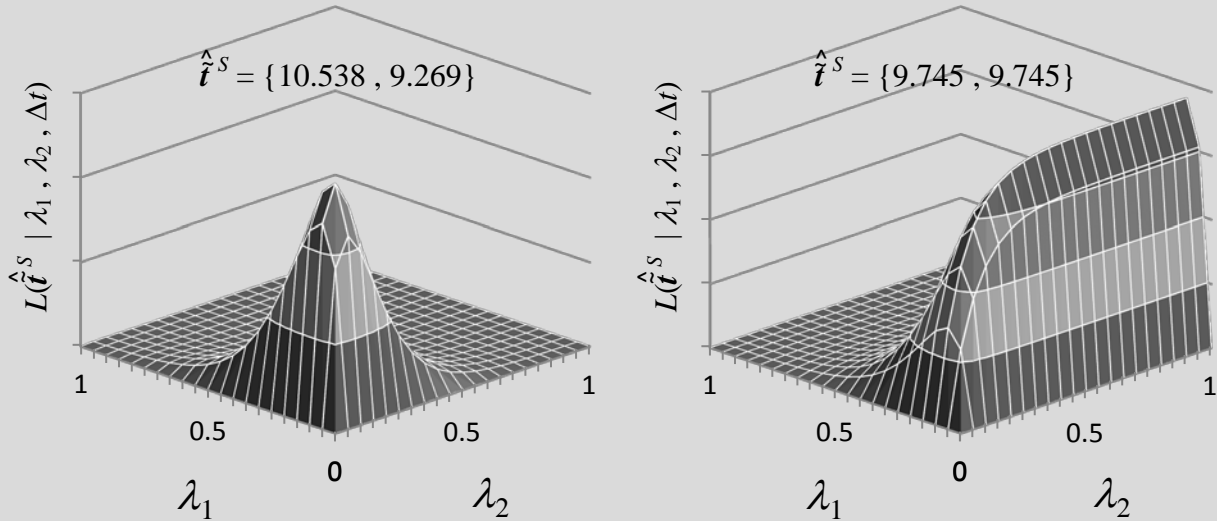


Figure 48: Likelihood functions of system illustrated in Figure 44 with evidence sets (151) and (155), analysed with an uncertainty in measurement of 0,001.

It can be seen that the limiting case of the general likelihood function in equation (184) when Δt approaches zero is equivalent to the likelihood functions from Example 18 and Example 19.

$$L(\hat{\mathbf{t}}^S | \boldsymbol{\theta}, \Delta t) = \begin{cases} 0 & \text{if } \hat{t}_1^S < \hat{t}_2^S \\ dt \lambda_1 e^{-\lambda_1 \hat{t}_2^S} (1 - e^{-\lambda_2 \hat{t}_1^S}) & \text{if } \hat{t}_1^S = \hat{t}_2^S \\ dt^2 \lambda_1 e^{-\lambda_1 \hat{t}_2^S} \lambda_2 e^{-\lambda_2 \hat{t}_1^S} & \text{if } \hat{t}_1^S > \hat{t}_2^S \end{cases} \quad \dots \text{noting that } \lim_{\Delta t \rightarrow 0} \frac{1 - e^{-x \Delta t}}{\Delta t} = x \quad \text{---(186)}$$

Uncertain data for continuous time based systems can be analysed using Berkson or Classical error in either equations (171), (172) or (173). However, a specific case of uncertain data analysis has been explored in the case of inherent time measurement inaccuracies, resulting in a simplified likelihood function in equation (178) which is demonstrated in Example 21. Recalling:

$$L(\hat{\mathbf{t}}^S | \boldsymbol{\theta}, \Delta t) = \prod_{i=1}^m \Pr(\hat{t}_i^S \leq T_i^S < (\hat{t}_i^S + \Delta t) | \tilde{\boldsymbol{\theta}}_i^{\subset S}, \hat{\mathbf{t}}_i^{\subset S}) \quad \text{---(178)}$$

It can be seen that equation (178) is in fact equivalent to the *likelihood in terms of observation method* written in equation (173), where the observed evidence \hat{E} is equivalent to a set of failure detection times and the inherent inaccuracy: $\hat{E} = \hat{\mathbf{t}}$.

$$\begin{aligned} L(\hat{\mathbf{t}}^S | \boldsymbol{\theta}, \Delta t) &= \prod_{i=1}^m \Pr(\hat{t}_i^S \leq T_i^S < (\hat{t}_i^S + \Delta t) | \tilde{\boldsymbol{\theta}}_i^{\subset S}, \hat{\mathbf{t}}_i^{\subset S}) \\ &\equiv \prod_{i=1}^m \int_{\hat{t}_i^S}^{\hat{t}_i^S + \Delta t} \Pr(T_i^S = t_i^S | \tilde{\boldsymbol{\theta}}_i^{\subset S}, \hat{\mathbf{t}}_i^{\subset S}) dt_i^S \end{aligned}$$

$$= \prod_{i=1}^m \int_{\forall t_i^S} \left[\Pr(\hat{t}_i^S | t_i^S, \Delta t) \Pr(T_i^S = t_i^S | \tilde{\theta}_i^{\subset S}, \hat{t}_i^{\subset S}) \right] dt_i^S$$

$$\text{where } \Pr(\hat{t}_i^S | t_i^S, \Delta t) = \begin{cases} 1 & \dots & \hat{t}_i^S < t_i^S \leq (\hat{t}_i^S + \Delta t) \\ 0 & \dots & \text{otherwise} \end{cases}, \text{ which is analogous to}$$

$$\text{classical error } \Pr(\hat{E} | E),$$

$$= \prod_{i=1}^m \left[\lim_{\Delta t_i^S \rightarrow 0} \left(\sum_{\hat{t}_i^S = 0, \Delta t_i^S, 2\Delta t_i^S, \dots}^{\infty} \left(\Pr(\hat{t}_i^S | t_i^S, \Delta t) \Pr(T_i^S = t_i^S | \tilde{\theta}_i^{\subset S}, \hat{t}_i^{\subset S}) \right) \right) \right]$$

$$= \prod_{i=1}^m \left[\lim_{\Delta t_i^S \rightarrow 0} \left(\sum_{\hat{t}_i^S = 0, \Delta t_i^S, 2\Delta t_i^S, \dots}^{\infty} \left(\Pr(\hat{E} | E, \Delta t) L(E | \tilde{\theta}_i^{\subset S}, \hat{t}_i^{\subset S}) \right) \right) \right]$$

which is simply products of the likelihood function proposed by Tan and Xi in equation (173).

7.5. SUMMARY

The likelihood functions developed in previous chapters for both on-demand and continuous time based systems are generated from first principles and inherently deal with uncertainties contained with various evidence states. The derivation of each likelihood function is analogous with the *likelihood in terms of observation method* developed by Tan and Xi. Additionally, the inherent inaccuracies that exist in time measurement devices were exploited for the case of continuous time-based systems to generate a computationally efficient likelihood function. This reduces computational time significantly, and reflects time measurement in reality.

Chapter 8: Sensor Placement: Maximising information from Bayesian analysis of complex systems

8.1. INTRODUCTION

At the heart of Bayesian analysis is the concept of improving state of knowledge and information. This is formally implemented by observing a process or system, gathering evidence and incorporating it through Bayes' Theorem to modify a joint distribution of the 'unknowns of interest', representing the state of knowledge. Typically, improving information through Bayesian analysis is in itself the only desired outcome, but in some instances it may be beneficial to quantify the amount of information 'improvement' in a way that allows the engineer to assess the nature of information gathering for the purpose of improving it. Alternately, the form or nature that evidence is gathered may have constraints (such as resource, time or physical) where it may be desirable to assess the probable improvement of information. This can allow different forms of evidence to be compared against each other for the probable gains in information.

Such an example for assessing the information gains from various forms of evidence gathering is posed by of *overlapping data sets* gathered from complex systems, as these sets are drawn from sensors whose placement may have some flexibility. An approach is developed in this chapter that allows the reliability engineer to optimise sensor placement in complex systems.

8.2. BAYESIAN EXPERIMENTAL DESIGN

Bayesian Experimental Design provides a framework through which experiments can be assessed against the expected value of a *utility function*. The *utility function* defines the ‘worth’ or ‘value’ of a particular experiment in a theoretical probabilistic Bayesian construct. Bayesian inference allows the *utility function* to incorporate both the meaning or information contained by observing a given set of evidence and any prior information that exists about the parameters or the unknowns of interest. [28]

The term ‘experiment’ typically conjures ideas of expeditionary activities representing an exploration to improve overall state of knowledge of the system or process in question. It implies that a ‘test environment’ is established along with its own variables and constraints in order to gain information about relevant parameters. Thus, an experiment differs fundamentally from actual use, operation or natural occurrence of the process or system as the latter instances do not involve a contrived ‘test environment’. Whilst this may be a discussion on semantics, the problem posed by the question of employing sensors within a system to improve the value of the state of knowledge can be considered to exist within the realms of both experimentation and actual use. The reason being is that the system function within its operational environment is independent of sensor placement, and sensor placement can be completely controlled by the test engineer.

Design of experiments historically focuses on the duration, size and conditions of test. However, in the analysis of systems, the ability to move and locate sensors at various levels within system

hierarchy is not limited to the system design phase, and sensors can continue to be placed in optimal locations during operational use of the system without affecting overall functionality. In this way, experimentation can be considered (from certain points of view) to continue after the completion of design phases. Ultimately systems can always be changed or modified for certain resource investments, so the ability to influence designs never disappears.

Recalling that Bayes' theorem is written formally as:

$$\pi_1(\boldsymbol{\theta} | E) = \frac{L(E | \boldsymbol{\theta}) \pi_0(\boldsymbol{\theta})}{\int_{\forall \boldsymbol{\theta}'} L(E | \boldsymbol{\theta}') \pi_0(\boldsymbol{\theta}') d\boldsymbol{\theta}'} \quad \text{---(1)}$$

where $\boldsymbol{\theta}$ is the set of *unknowns of interest* or *parameters*, $\pi_0(\boldsymbol{\theta})$ is the prior distribution of $\boldsymbol{\theta}$ representing the initial state of knowledge, $L(E | \boldsymbol{\theta})$ is the likelihood of observing a set of evidence, E , for a given $\boldsymbol{\theta}$, and $\pi_1(\boldsymbol{\theta} | E)$ is the updated posterior distribution of the set of *unknowns of interest* or parameters representing the updated state of knowledge.

The nature of the evidence is inherently linked to the nature of the experiment and physical rules of observation. For example, if an experiment, ε_1 , involves testing the reliability of 10 components and recording failure times, then the evidence set, E_1 , consists of failure times $\{t_1, t_2, \dots, t_{10}\}$. If another experiment, ε_2 , involves testing the same 10 components but only observing their state after a test duration, T , then the evidence set, E_2 , consists of the number of failed and surviving components at time T , $\{F_T, S_T, T\}$, noting that $F_T + S_T = 10$. Each experiment is observing the same process, but is structured differently and thus implies different

evidence sets. This is an often assumed aspect of the likelihood function, $L(E | \theta)$ and the experimental framework is rarely written explicitly.

When this aspect of the likelihood function is not assumed and the experimental framework is denoted ε , then Bayes' Theorem in equation (1) is more completely written as:

$$\pi_1(\theta | E, \varepsilon) = \frac{L(E | \theta, \varepsilon) \pi_0(\theta)}{\int_{\forall \theta} L(E | \theta', \varepsilon) \pi_0(\theta') d\theta'} \quad \text{---(187)}$$

where ε is the framework (experimental or otherwise) through which the evidence E is gathered.

The likelihood function in equation (187) can be written as either $L(E | \theta, \varepsilon)$ or $L_\varepsilon(E | \theta)$. The discussion below follows steps extensively covered in literature on the topic such as Lindley [29]. The question that then arises is: *‘What experimental framework, ε , should be established in order to optimize the “value” of the experiment?’* This can be achieved by considering the probability of observing specific evidence sets based on the prior information of the parameter set θ . For example, if there is no uncertainty with the parameter set θ , (implying that all parameters are known), then it can be written

$$\Pr(E | \theta, \varepsilon) = \text{the probability of observing evidence set } E \quad \text{---(188)}$$

for parameter set θ in experimental framework ε .

However, should these parameters be known with absolute certainty then there is no need to conduct an experiment. In reality, there is by definition uncertainty in the parameter set θ which is summarized by the prior distribution $\pi_0(\theta)$. The probability of observing a given set of evidence becomes marginalized as it is a function of $\pi_0(\theta)$ and the following can be written:

$$\Pr[E | \varepsilon, \pi_0(\theta)] \text{ or } \Pr(E | \varepsilon, \pi_0) = \int_{\forall \theta} \Pr(E | \varepsilon, \theta') \pi_0(\theta') d\theta' \quad \text{---(189)}$$

noting that the symbol ' π ' when used throughout this dissertation in isolation represent a joint probability density function of a set of unknowns of interest or parameters.

Equation (189) allows the probability of observing specific instances of the evidence set E based on the experimental framework ε in the information context provided by prior information.

8.3. UTILITY FUNCTION

The utility function quantifies the 'value' of an experiment to the test engineer, and is designated with the symbol U . When the utility function is based on uncertainty or information, it is by definition derived from the nature of the 'state of knowledge' of the unknowns of interest. Therefore, *information utility*, U^I , must be a function of the joint probability density function of all relevant unknowns of interest or parameters.

$$\text{i.e. Information utility} = U^I[\pi(\theta)] \text{ or } U^I(\pi) \quad \text{---(190)}$$

When considering the information utility associated with an observed set of evidence, E , within an experimental framework, ε , the calculation is still derived from joint probability density functions. It can be based on the posterior distribution, $\pi_1(\boldsymbol{\theta} | E, \varepsilon)$, such as the information or uncertainty implied with prior knowledge, E and ε . Alternately, the information utility can be a comparison between the utilities of the posterior distribution and the prior distribution $\pi_0(\boldsymbol{\theta})$.

i.e. Information utility of E and ε

$$= U^I [E, \varepsilon | \pi_0(\boldsymbol{\theta})] \text{ or } U^I (E, \varepsilon | \pi_0) \text{ or } U^I (\pi_1 | \pi_0) \quad \text{---(191)}$$

noting that E and ε given π_0 defines π_1 as per Bayes' theorem in equation (187).

The evidence set, E , is resultant from random processes, and thus is outside the control of the test engineer. However, the experimental framework, ε , is completely controlled by the test engineer, and in the context of complex systems consists of various arrangements of sensors within the system logic. It is therefore useful to attribute information utility to the experimental framework only. In the case of on-demand systems, there is only a set of discrete possible evidence sets, \hat{E}_j , meaning the information utility of an experimental framework is described by a discrete probability density function.

$$\text{i.e. } f(U_j^I) = \Pr(U_j^I) = \Pr(\hat{E}_j | \varepsilon, \pi_0) \quad \text{---(192)}$$

where U_j^I is $U_j^I(\hat{E}_j, \varepsilon | \pi_0)$, the information utility of the j^{th} possible evidence set, \hat{E}_j , and the experimental framework, ε , given the prior distribution $\pi_0(\boldsymbol{\theta})$.

The expected information utility, \bar{U}^I , can then be calculated by finding the mean of the distribution in equation (192):

$$\bar{U}^I(\varepsilon | \pi_0) = \int_{\forall j} U_j^I \bullet f(U_j^I) dU_j^I = \sum_{\forall j} U_j^I \bullet \Pr(\hat{E}_j | \varepsilon, \pi_0) \quad \text{---(193)}$$

Substituting equation (189) into equation (193) yields equation (194).

$$\bar{U}^I(\varepsilon | \pi_0) = \sum_{\forall j} U_j^I \bullet \left(\int_{\forall \theta} \Pr(\hat{E}_j | \varepsilon, \theta') \pi_0(\theta') d\theta' \right) \quad \text{---(194)}$$

If the amount of information ‘improvement’ between prior and posterior distributions is valuable, then $\bar{U}^I(\varepsilon | \pi_0)$ would be constructed in a manner that it is maximized when this is achieved. The experimental framework with the highest expected utility then becomes the most ‘valuable’ to the test engineer allowing optimization to occur. In some instances, it is possible for the utility of an experimental framework to be superior to all others for all possible prior distributions meaning that it is the optimal framework in any possible scenario [29].

8.4. INFORMATION OPTIMIZATION THROUGH INFORMATION UTILITY FUNCTIONS

There is significant literature regarding the design of experiments to maximize information of parameters. Information can be viewed as the inverse of uncertainty. Typically, information is characterized primarily on the scaled or normalized magnitude of the variance of parameters and

is considered optimal when the smallest possible variance is achieved. Two popular information metrics are listed below.

8.4.1. Fisher Information.

Explored by Fisher, the *Fisher Information* of a set of random variables with a joint probability distribution provides the lower bounds on variance and covariance. Maximizing *Fisher Information* therefore minimizes the variation of these random variables [30].

8.4.2. Shannon Information.

Developed by Shannon [31], *Shannon Information*, described by *Differential Entropy* quantifies the uncertainty of a set of random variables with a joint probability distribution. Minimizing the *differential entropy* decreases the uncertainty or disorder, and hence increases information.

Shannon introduced the concept of differential entropy in an influential paper which has been the basis of much subsequent analysis in the field of information theory [31]. Entropy is a measure of uncertainty associated with a random variable. Entropy of a discrete random variable is a non-negative number where 0 represents total certainty, and is defined by:

$$H(\boldsymbol{\theta}) = -\sum_{\forall \boldsymbol{\theta}} p(\boldsymbol{\theta}) \log p(\boldsymbol{\theta}) \quad \text{---(195)}$$

where $H(\boldsymbol{\theta})$ is the entropy of the discrete random variable set $\boldsymbol{\theta}$, and $p(\boldsymbol{\theta})$ is the probability of $\boldsymbol{\theta}$.

Entropy of a continuous random variable is more difficult to estimate as the limiting case of an increasingly discretized continuous probability distribution is always 0. *Differential entropy* of a continuous random variable is analogous to the entropy of a discrete random variable, but has several unique properties. For example, the differential entropy can have a negative value, and increasingly negative values represent increasing certainty.

$$h(\boldsymbol{\theta}) = - \int_{\forall \boldsymbol{\theta}} f(\boldsymbol{\theta}) \log f(\boldsymbol{\theta}) d\boldsymbol{\theta} \quad \text{---(196)}$$

where $h(\boldsymbol{\theta})$ is the *differential entropy* of the continuous random variable set $\boldsymbol{\theta}$, and $f(\boldsymbol{\theta})$ is the probability density function of $\boldsymbol{\theta}$.

The *differential entropy* defined in equation (196) can be estimated using techniques described in [32]. This metric can then be used to assess the uncertainty (and hence information) associated with a posterior distribution defined by Bayes' Theorem. The units associated with *differential entropy* are dependent on the base of the logarithm in equation (196). The units are 'nats' when the natural logarithm is applied and will be the units used throughout this dissertation.

A continuous random variable that involves less uncertainty (and hence more information) will have a lower differential entropy and can be used to assess the information associated with a posterior distribution defined by Bayes' Theorem. The concept of marginal differential entropy will be introduced and used throughout this dissertation to assess the amount information a joint posterior distribution has about a particular parameter or sub-set of parameters (it is based on the entropy of marginal posterior distributions of these sub-sets).

$$\begin{aligned}
h_m(\boldsymbol{\theta}^\bullet) &= - \int_{\forall \boldsymbol{\theta}^\bullet} f_m(\boldsymbol{\theta}^{\bullet'}) \log f_m(\boldsymbol{\theta}^{\bullet'}) d\boldsymbol{\theta}^{\bullet'} \\
&= - \int_{\forall \boldsymbol{\theta}^\bullet} f(\boldsymbol{\theta}^{\bullet'}) \log \left(\int_{\forall \boldsymbol{\theta}^\circ} f(\boldsymbol{\theta}^\circ; \boldsymbol{\theta}^{\bullet'}) d\boldsymbol{\theta}^\circ \right) d\boldsymbol{\theta}^{\bullet'} \quad \text{---(197)}
\end{aligned}$$

where $\boldsymbol{\theta}^\bullet$ is the sub-set of continuous random variables that are being investigated, $\boldsymbol{\theta}^\circ$ is the remaining sub-set of continuous random variables which are *not* being investigated (making $\boldsymbol{\theta}^\circ$ the complement of $\boldsymbol{\theta}^\bullet$ and therefore $\boldsymbol{\theta} = \{\boldsymbol{\theta}^\bullet, \boldsymbol{\theta}^\circ\}$), $h_m(\boldsymbol{\theta}^\bullet)$ is the *marginal differential entropy* of the sub-set of continuous random variable $\boldsymbol{\theta}$, and $f(\boldsymbol{\theta})$ is the probability density function of $\boldsymbol{\theta}$.

8.5. INFORMATION UTILITY FUNCTION

There are other information metrics within literature that have not been mentioned at this point as it is not the intent of this dissertation to advocate the use of any metric in particular. In any case, the test engineer conducts experimentation in order to gain information to make physical predictions with certain levels of confidence. The requisite levels of *Fisher* or *Shannon Information* for the relevant parameters to achieve this level of confidence would not be directly calculable. It may be that the confidence on the physical prediction represents the utility of the experimental framework. Using *Fisher* or *Shannon Information* to optimize experimental framework utility may be used by the test engineer as it is assumed it will improve the confidence of subsequent physical predictions, making the utility a subjective measurement. Optimization techniques focus on specific aspects of information metrics employed at the discretion of the test engineer or data analyst.

Once a relevant information metric has been determined, it then becomes the core of the utility function of a given evidence framework, ε . However, the following aspects of the utility function need to be considered:

8.5.1. Information of the Posterior Distribution.

The information of the *posterior distribution* represents the resultant information of the experiment, which involves both the prior information and the information gained by the experiment.

8.5.2. Information difference of Posterior/Prior distributions.

Much literature focuses on the *information difference of posterior/prior distributions*, as this represents the information gained by the experimental framework, ε . Lindley discusses this at length in regard to the information gain in *Shannon Information* throughout [29].

8.5.3. Information of specific parameters.

It may be the case that information of certain parameters is more valuable than others or need to be treated differently. For instance, it may be desirable to optimize the minimum information of the parameter set (i.e. improve the information of the parameter with the least prior information about it). It may be that the test involves a system or process with many parameters, but one parameter in particular is the most valuable in terms of the confidence of subsequent physical predictions.

8.5.4. Information of variables that are functions of parameters.

The physical relevance of parameters is once again, subjective. For example, it may be desirable to gain information about the time to failure of a particular component. This time to failure is itself a function of parameters and this may be the most relevant metric that is being investigated.

The utility function can be determined based on these factors above. Several examples of information based utility functions for the purpose of optimizing experimental frameworks are included in Table 21.

| Utility Function | Description |
|--|---|
| $U^I(\varepsilon, E \pi_0) = \begin{cases} \int_{\forall \theta} \ln[\pi_1(\theta' E', \varepsilon)] \pi_1(\theta' E', \varepsilon) d\theta' \\ - \int_{\forall \theta} \ln[\pi_0(\theta')] \pi_0(\theta') d\theta' \end{cases}$ | Expected improvement of <i>Shannon Information</i> by posterior distribution when compared to prior distribution [29] |
| $U^I(\varepsilon, E \pi_0) = \min\left(\frac{1}{\sigma_{\theta}^2}\right) \dots \text{ where } \sigma_{\theta}^2 = \{\sigma_{\theta_1}^2, \sigma_{\theta_2}^2, \dots, \sigma_{\theta_n}^2\} \text{ and } \theta \sim \pi_1(\theta E, \varepsilon)$ | Inverse of the largest posterior parameter variance |
| $U^I(\varepsilon, E \pi_0) = \frac{1}{\sum_{j=1}^n \sigma_{\theta_j}^2} \dots \text{ where } \theta \sim \pi_1(\theta E, \varepsilon)$ | Inverse of the sum of the posterior variance of all parameters. |

Table 21: Examples of information based *Utility Functions* of experimental frameworks

8.6. SENSOR PLACEMENT

Placing sensors within systems comes at sometimes significant resource costs. Physical limitations (such as volume and temperature) may impose constraints on the number and locations of sensors. It is suggested that the issue of sensor placement be considered a holistic multi-objective optimization problem, where the information utility is but one objective of many. Many techniques exist for multi-objective optimization (see Steuer [33]) but they will not be explored herein.

The placement of sensors based on information optimality is necessarily complicated by many factors, each of which is addressed in the following proposed steps to optimize sensor placement for maximal information.

8.6.1. Prior Information.

The prior information of components affects information that sensors yield about all other components demonstrating the need for utility functions to be dependent on prior information in equations above. For example, if a basic two component series system comprising of components A and B has a system level sensor that detects high systemic failure probability and prior information suggesting that component A is very reliable, then the sensor evidence infers that component B is very unreliable. Conversely, if the prior information suggest component A is very unreliable, then minimal information is yielded about the reliability characteristics of component B as systemic level failure will most likely be caused by component A failure.

8.6.2. Available Sensor Locations.

Some systems cannot allow sensor to be located at all hierarchical positions within system logic. Potential sensor sites are typically limited through physical and environmental constraints. Whether part of a multi-objective optimization problem or otherwise, all possible sensor locations need to be identified. Wherever possible, the set ϵ should be minimized through the realization of any physical constraints of the system to limit computational time.

$$\epsilon = \{\epsilon_1, \epsilon_2, \epsilon_3, \dots\} \text{ where } \epsilon_i \text{ is a particular permutation of sensor locations.} \quad \text{---(198)}$$

8.6.3. Information Utility.

The *utility function* needs to be selected, and could be any of the examples listed in Table 21. The utility function will be a condition function of the posterior distribution given the prior distribution of the unknowns of interest.

$$U^I(E, \epsilon | \pi_0) \text{ or } U^I(\pi_1 | \pi_0) \quad \text{---(199)}$$

8.6.4. System Logic (Bayesian Analysis).

If a sensor is not immediately ‘above’ a component when represented hierarchically, the system logic is required to generate inference about subordinate components. The amount of information decreases based on the complexity and nature of the system logic.

8.6.5. Nature of the evidence.

The evidence set can affect the amount of information gathered. The nature includes the size of the evidence set and the observation methodology. For example, should the sensors detect failure, then they may either record the time at which failure was detected or allow the test engineer to identify the failure of sub-systems on routine inspections. The first framework will yield exact failure times ($T = t$), while the latter will yield upper limits on failure times ($T \leq /> t$). For on-demand systems, evidence will always be of the form of k failures from r demands or equivalent. To assess the expected information utility of sensor placements, the number of demands, r , needs to be assumed. An outcome of an analysis of the nature of the evidence is naturally the likelihood function, $L(E | \theta, \varepsilon)$.

8.6.6. Deriving structure functions.

The state detected by each sensor is a function of the component state vector, $\tilde{\mathbf{x}}$, which is a vector that contains the state of each component in the system. The output of the structure is the *sensor information vector*, $\tilde{\mathbf{x}}^S$.

$$\text{i.e. } \tilde{\mathbf{x}}^S = \{x_1^S, x_2^S, \dots, x_i^S, \dots, x_m^S\} = \tilde{\boldsymbol{\phi}}^S(\tilde{\mathbf{x}}) \quad \text{---(68)}$$

where $\tilde{\boldsymbol{\phi}}^S$ is the vector of the structure functions for all sensors, $\{\phi_1^S, \phi_2^S, \dots, \phi_i^S, \dots, \phi_m^S\}$.

8.6.7. Simulation of evidence for on-demand systems.

The possible evidence sets along with their probabilities need to be simulated from the prior distribution of unknowns of interest, $\pi_0(\boldsymbol{\theta})$. There are multiple approaches to this, but an approach utilizing Monte-Carlo simulation is detailed below.

- a. *Sampling unknowns of interest – $\boldsymbol{\theta}$.* Monte-Carlo simulation can be used to randomly draw joint samples of the unknowns of interest.
- b. *Simulation of component state vectors – $\hat{\mathbf{x}}$.* Each randomly drawn set of the *unknowns of interest* will define component state probabilities. Monte-Carlo simulation can then be used to randomly draw *component state vectors*. The resultant set of *component state vectors* is shown in equation (200).

$$\hat{\mathbf{x}} = \{\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \hat{\mathbf{x}}_3, \dots, \hat{\mathbf{x}}_d\} \quad \text{where } d \text{ is the number of samples.} \quad \text{---(200)}$$

noting that $\hat{\mathbf{x}}$ will probably contain multiple instances of the same *component state vector*.

This set allows probabilities for all component state vectors to be estimated.

$$\Pr(\hat{\mathbf{x}} | \pi_0) \approx \frac{\text{number of times } \hat{\mathbf{x}} \text{ occurs in } \hat{\mathbf{x}}}{d} \quad \text{---(201)}$$

- c. *Probabilities of sensor information vectors* - $\hat{\mathbf{x}}^S$. Each component state vector, $\hat{\mathbf{x}}$, will define a *sensor information vector* as shown in equation (68). The probability of each simulated *sensor information vector* is approximated in equation (202).

$$\Pr(\hat{\mathbf{x}}_l^S | \varepsilon, \pi_0) \approx \sum_{\forall \hat{\mathbf{x}} (\in \hat{\mathbf{x}}) \rightarrow \hat{\mathbf{x}}_l^S} \Pr(\hat{\mathbf{x}} | \pi_0) \quad \text{---(202)}$$

- d. *Sampling combinations of sensor information vectors* - $\hat{\mathbf{v}}^S$. Monte-Carlo simulation using the probabilities for *sensor information vectors* given in (202) can be used to randomly draw combinations of *sensor information vectors*. The number of *sensor information vectors* in each combination is r , an assumed number of demands.

$$\hat{\mathbf{v}}^S = \{\hat{\mathbf{v}}_1^S, \hat{\mathbf{v}}_2^S, \hat{\mathbf{v}}_3^S, \dots, \hat{\mathbf{v}}_d^S\} \quad \text{---(203)}$$

where d is the number of samples, and each *combination of sensor information vectors*, $\hat{\mathbf{v}}^S$, is defined in equation (204), noting that $\hat{\mathbf{v}}^S$ will probably contain multiple instances of the same *combination of sensor information vectors*.

$$\hat{\mathbf{v}}^S = \{\hat{\mathbf{v}}_1^S, \hat{\mathbf{v}}_2^S, \hat{\mathbf{v}}_3^S, \dots\} \quad \text{---(204)}$$

where $\hat{\mathbf{v}}_l^S$ is the number of times the *sensor information vector* $\hat{\mathbf{x}}_l^S$, appears in the combination $\hat{\mathbf{v}}^S$ noting that $\hat{\mathbf{v}}_1^S + \hat{\mathbf{v}}_2^S + \hat{\mathbf{v}}_3^S + \dots = r$.

The probability of each combination of *sensor information vectors*, $\hat{\mathbf{v}}^S$, occurring can be approximated from equation (203).

$$\Pr(\hat{\mathbf{v}}^S | \varepsilon, \pi_0) \approx \frac{\text{number of times } \hat{\mathbf{v}}^S \text{ occurs in } \hat{\mathbf{v}}}{d} \quad \text{---(205)}$$

e. *Simulation of evidence sets.* Each combination, $\hat{\mathbf{v}}^S$, implies a particular evidence set, \hat{E} .

$$\hat{E}|_{\hat{\mathbf{v}}^S} = \left\{ \hat{\mathbf{k}}_1^S, \hat{\mathbf{k}}_2^S, \hat{\mathbf{k}}_3^S, \dots, \hat{\mathbf{k}}_i^S, \dots, \hat{\mathbf{k}}_m^S \right\}_{\hat{\mathbf{v}}^S} \quad \text{---(206)}$$

where $\hat{E}|_{\hat{\mathbf{v}}^S}$ is the evidence set implied by the combination of *sensor information vectors*, $\hat{\mathbf{v}}^S$;

$$\hat{\mathbf{k}}_i^S|_{\hat{\mathbf{v}}^S} = \left\{ \hat{k}_i^{S(1)}, \hat{k}_i^{S(2)}, \hat{k}_i^{S(3)}, \dots, \hat{k}_i^{S(z-1)} \right\}_{\hat{\mathbf{v}}^S} \quad \text{---(207)}$$

$\hat{\mathbf{k}}_i^S|_{\hat{\mathbf{v}}^S}$ is the vector of states detected by the i^{th} sensor implied by the combination of *sensor information vectors*, $\hat{\mathbf{v}}^S$; and

$$\hat{k}_i^{S(x_i^S)}|_{\hat{\mathbf{v}}^S} = \sum_{l=1,2,3,\dots} \begin{cases} \hat{v}_l^S \dots \text{if } (\hat{\mathbf{x}}_l^S)_i = x_i^S \\ 0 \dots \text{otherwise} \end{cases} \quad \text{---(208)}$$

where $(\hat{\mathbf{x}}_l^S)_i$ is the i^{th} element of $\hat{\mathbf{x}}_l^S$, or the state implied by i^{th} sensor in the sensor information vector, $\hat{\mathbf{x}}_l^S$.

At this stage, the probabilities of all combinations of *sensor information vectors*, $\Pr(\hat{\mathbf{v}}^S | \varepsilon, \pi_0)$, have been approximated, and each possible evidence set, \hat{E} , can be calculated as a function of $\hat{\mathbf{v}}^S$ through equations (206), (207) and (208). This allows the probability of possible evidence sets to be calculated as shown in equation (209).

$$\Pr(\hat{E}_j | \varepsilon, \pi_0) \approx \sum_{\forall \hat{\mathbf{v}}^S (\in \hat{\mathbf{V}}^S) \rightarrow \hat{E}_j} \Pr(\hat{\mathbf{v}}^S | \varepsilon, \pi_0) \quad \text{---(209)}$$

where \hat{E}_j is the j^{th} evidence set that is permissible from the entire set of simulated combinations of *sensor information vectors*, $\hat{\mathbf{V}}^S$.

8.6.8. Simulation of evidence for continuous time based systems.

The steps for continuous time based systems are analogous to those explored above for on-demand systems.

- a. *Sampling unknowns of interest – θ* . Monte-Carlo simulation can be used to randomly draw joint samples of the unknowns of interest.
- b. *Simulation of component failure times – \hat{t}* . Each randomly drawn set of the *unknowns of interest* will define the time to failure probability distributions for each component. Monte-Carlo simulation can then be used to randomly draw component failure times using the inverse (or approximate inverse) of each component's *CDF*. These simulations will yield exact times, but need to be rounded down to the nearest multiple of the pre-selected

measurement accuracy, $\Delta \hat{t}$. For example, for a system with three components, randomly drawn times to failure {1.78462... , 9.34289... , 10.32791...} become {1.78 , 9.34 , 10.33} for a measurement accuracy of $\Delta \hat{t} = 0.01$. This measurement accuracy can subjectively be made larger to limit the computational resources required in subsequent steps. The resultant set of *component failure time vectors* is shown in equation (210).

$$\hat{\mathbf{t}} = \{\hat{t}_1, \hat{t}_2, \hat{t}_3, \dots, \hat{t}_d\} \quad \text{where } d \text{ is the number of samples.} \quad \text{---(210)}$$

noting that $\hat{\mathbf{t}}$ will probably contain multiple instances of the same *component failure time vector*.

This set allows probabilities for all component state vectors to be estimated.

$$\Pr(\hat{\mathbf{t}} | \pi_0) \approx \frac{\text{number of times } \hat{\mathbf{t}} \text{ occurs in } \hat{\mathbf{t}}}{d} \quad \text{---(211)}$$

- c. *Probabilities of time to sensor failure detection vectors - $\hat{\mathbf{t}}^S$* . Each *component failure time vector*, $\hat{\mathbf{t}}$, will define a *time to sensor failure detection vectors* $\hat{\mathbf{t}}^S$ based on system logic such as that represented in structure functions in equation (68). The probability of each simulated *sensor information vector* is approximated in equation (212).

$$\Pr(\hat{E} | \varepsilon, \pi_0) = \Pr(\hat{\mathbf{t}}^S | \varepsilon, \pi_0) \approx \sum_{\forall \hat{\mathbf{t}} \in \hat{\mathbf{t}} \rightarrow \hat{\mathbf{t}}^S} \Pr(\hat{\mathbf{t}} | \pi_0) \quad \text{---(212)}$$

noting that the simulated evidence set, \hat{E} , is simply $\hat{\mathbf{t}}^S$.

8.6.9. Simulation of posterior distributions.

As discussed previously, the *information utility* of each simulated evidence set requires the posterior distribution of the unknowns of interest to be calculated. Each simulated posterior distribution is defined by Bayes' Theorem:

$$\hat{\pi}_1^a(\boldsymbol{\theta} | \hat{E}_j, \varepsilon) \approx \frac{L(\hat{E}_j | \boldsymbol{\theta}, \varepsilon) \pi_0(\boldsymbol{\theta})}{\int_{\forall \boldsymbol{\theta}} L(\hat{E}_j | \boldsymbol{\theta}', \varepsilon) \pi_0(\boldsymbol{\theta}') d\boldsymbol{\theta}'} \quad \text{---(213)}$$

8.6.10. Expected Information Utility.

The *information utility* function in equation (199) along with the probability and posterior distribution of each evidence set, equations (209) and (213) respectively can then be substituted into equation (193) to derive the expected *information utility* of the experimental framework, ε (i.e. the sensor placement).

Example 22. Expected information utility for various sensor placement arrangements for an on-demand system.

Consider the binary-state on-demand system illustrated in Figure 49. A more detailed understanding of the *unknowns of interest* (the component failure probabilities p_1 , p_2 and p_3) is desired but the placement of sensors within the system involves some costs. The system is always monitored at the ‘top-event’ level (i.e. systemic failure is always detected on occurrence) and there is scope to place additional sensors at the locations denoted ‘Possible Sensor #2’ and ‘Possible Sensor #3’ in Figure 49.

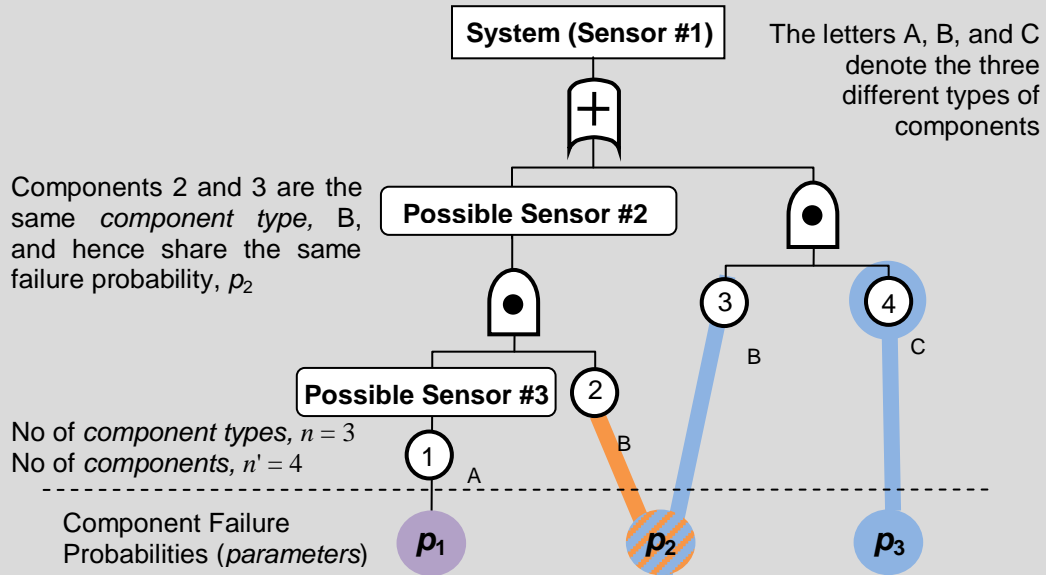


Figure 49: 4 component (3 component type) binary-state on-demand system

In this instance, prior uniform distributions are assumed for all component failure probabilities.

$$\text{i.e. } \pi_0(\theta) = \pi_0(p_1, p_2, p_3) = \prod_{i=1}^3 \pi_0(p_i) \quad \text{where } \pi_0(p_i) = \begin{cases} 1, & 0 \leq p_i \leq 1 \\ 0, & \text{otherwise} \end{cases} \quad \text{---(214)}$$

The information utility used to assess the value of various sensor placements is selected to be the inverse of the sum of the posterior variance of all parameters (row three of Table 21) is developed in equation (215).

$$U^I(\varepsilon, E | \pi_0) = U^I(\pi_1 | \pi_0) = \frac{1}{\sum_{j=1}^n \sigma_{\theta_j}^2} \dots \text{ where } \boldsymbol{\theta} \sim \pi_1(\boldsymbol{\theta} | E, \varepsilon) \quad \text{---(215)}$$

The possible sensor arrangements are:

$$\varepsilon_1 = \{1\} \dots \text{ (i.e. systemic level failure detection only)} \quad \text{---(216)}$$

$$\varepsilon_2 = \{1,2\} \dots \text{ (i.e. systemic level and sensor \#2 failure detection)} \quad \text{---(217)}$$

$$\varepsilon_3 = \{1,3\} \dots \text{ (i.e. systemic level and sensor \#3 failure detection)} \quad \text{---(218)}$$

$$\varepsilon_4 = \{1,2,3\} \dots \text{ (i.e. systemic level, sensor \#2 and \#3 failure detection)} \quad \text{---(219)}$$

The likelihood function for this on-demand system is expressed in equation (220), noting that a full explanation and demonstration of the likelihood function is included in previous chapters.

$$L(E | \varepsilon, \mathbf{p}) \propto \sum_{\forall \tilde{\mathbf{v}}_a \in \mathbf{v}_E} \left[\prod_{l=1}^{2^4} \left\{ \frac{1}{(v_l)_a!} \left[\prod_{b=1}^4 p_{j_b}^{((x_b)_l)} (1 - p_{j_b})^{[1 - (x_b)_l]} \right]^{(v_l)_a} \right\} \right] \quad \text{---(220)}$$

where the unknown of interest, $\mathbf{p} = \{p_1, p_2, p_3\}$ is the set of 3 lower level *component type* failure probabilities, p_{j_b} is the failure probability of the b^{th} component (which is the failure probability of j_b^{th} *component type*), $\tilde{\mathbf{v}}_a$ is the a^{th} combination of r *state vectors* (each *state vector* comprises of n' component states), $(v_l)_a$ is the number of occurrences of the l^{th} *state*

vector in $\tilde{\mathbf{v}}_a$ and $(x_b)_l$ is the *state variable* of the b^{th} component in the l^{th} *state vector*.

For the purposes of this scenario, the sensor placement information utility will be base on $r = 5$ demands. The structure functions for each possible sensor location are based on where x_1, x_2, x_3 and x_4 (the state variables of all four components), and are shown in equations (221), (222), and (223).

$$\text{Sensor \#1: } \phi_1 = \phi_2 + x_3x_4 - \phi_2x_3x_4 \quad \text{---(221)}$$

$$\text{Sensor \#2: } \phi_2 = \phi_3x_2 \quad \text{---(222)}$$

$$\text{Sensor \#3: } \phi_3 = x_1 \quad \text{---(223)}$$

Using on Monte-Carlo simulation, the set $\{p_1, p_2, p_3\}$ is drawn a large number of times based on the prior distribution in equation (214). Each draw defines the state probabilities of each component, each draw equivalent to a component state vector, $\hat{\mathbf{x}}$. This process yields the component state vectors and their probabilities in Table 22.

| $\Pr(\hat{\mathbf{x}} \pi_0) \approx$ | | $\frac{1}{12}$ | $\frac{1}{12}$ | $\frac{1}{24}$ | $\frac{1}{24}$ | $\frac{1}{24}$ | $\frac{1}{24}$ | $\frac{1}{12}$ | $\frac{1}{12}$ | $\frac{1}{12}$ | $\frac{1}{12}$ | $\frac{1}{24}$ | $\frac{1}{24}$ | $\frac{1}{24}$ | $\frac{1}{24}$ | $\frac{1}{12}$ | $\frac{1}{12}$ |
|---|--------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| $\hat{\mathbf{x}}$ | Com 1: | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| | Com 2: | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| | Com 3: | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| | Com 4: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

Table 22: All possible *component state vectors* with probabilities of occurrence

The dependence between the states of components 2 and 3 (which are the same component type) is clearly seen in Table 22. All state vectors where components 2 and 3 have the same state (i.e. either both 0 or both 1) have a higher probability of occurring. All other state vectors where the states of components 2 and 3 are different have a lower probability of occurring. This means that component 2 is more likely to be in the same state as that of component 3 and vice versa. This stems from a common failure probability, p_2 .

Each *component state vector* generates a corresponding *sensor information vector*. Based on the data in Table 22, all possible *sensor information vectors* along with their probabilities can be calculated as shown in Table 23.

| $\Pr(\hat{\mathbf{x}}_l^S \varepsilon, \pi_0) \approx$ | | $\frac{3}{8}$ | $\frac{5}{24}$ | $\frac{1}{4}$ | $\frac{1}{8}$ | $\frac{1}{24}$ |
|--|-----------|---------------|----------------|---------------|---------------|----------------|
| $\hat{\mathbf{x}}^S$ | Sensor 1: | 0 | 0 | 1 | 1 | 1 |
| | Sensor 2: | 0 | 0 | 1 | 0 | 0 |
| | Sensor 3: | 0 | 1 | 1 | 0 | 1 |

Table 23: All possible *sensor information vectors* with probabilities of occurrence

The *sensor information vectors* in Table 23 are used for Monte-Carlo Simulation of evidence sets, \hat{E} . In this example, 100 000 simulations with 90 distinct evidence sets are generated. Each simulated evidence, set along with equation (220), can be substituted into equation (188) to yield a posterior distribution as shown in equation (213). The utility function for each posterior distribution is given in equation (215), which is simply the inverse of the sum of the variances of each unknown of interest. The ten most probable of these evidence sets when considering sensor

arrangement ε_4 (three sensors installed) are listed in Table 24, along with the utility of each.

| Evidence Set Number, j | Approximate probability of evidence set occurring, $\Pr(\hat{E}_j \varepsilon, \pi_0)$ | Evidence set, \hat{E}_j (based on $r = 5$ demands) | | | Utility, U_j^I |
|--------------------------|--|--|--|--|------------------|
| | | Number of observed failures by Sensor #1 | Number of observed failures by Sensor #2 | Number of observed failures by Sensor #3 | |
| 1 | 0.065918 | 2 | 1 | 2 | 22.061 |
| 2 | 0.054932 | 2 | 2 | 3 | 21.745 |
| 3 | 0.054932 | 1 | 1 | 2 | 20.090 |
| 4 | 0.048828 | 2 | 1 | 3 | 23.576 |
| 5 | 0.047607 | 3 | 2 | 3 | 23.856 |
| 6 | 0.045776 | 1 | 1 | 3 | 21.104 |
| 7 | 0.032959 | 2 | 1 | 1 | 21.972 |
| 8 | 0.032959 | 2 | 2 | 2 | 22.884 |
| 9 | 0.032959 | 3 | 2 | 2 | 23.800 |
| 10 | 0.032043 | 1 | 0 | 2 | 23.912 |
| ... | | | | | |

Table 24: Possible evidence sets with probabilities of occurrence (10 most probable of 90 evidence sets simulated)

Equation (193) can then be used to generate the expected utility of the sensor placement arrangement. This is demonstrated in equation (224) for sensor arrangement ε_4 .

$$\bar{U}^I(\varepsilon_1 | \pi_0) = \sum_{\forall j} U_j^I \bullet \Pr(E_j | \varepsilon, \pi_0) = \left\{ \begin{array}{l} 0.065918 \bullet 22.061 \\ +0.054932 \bullet 21.745 \\ +0.054932 \bullet 20.090 \\ \dots \end{array} \right\} = 23.746 \quad \text{---(224)}$$

The same process is used for all other sensor arrangements. The resultant expected utility for each is summarized in Table 25.

| ε_i | Expected Information Utility, $\bar{U}^I(\varepsilon_i \pi_0)$ |
|-----------------|--|
| none | 11.964 |
| {1} | 14.838 |
| {1,2} | 17.558 |
| {1,3} | 20.645 |
| {1,2,3} | 23.746 |

Table 25: Expected information utility for each sensor arrangement.

It can be seen that as expected, using all three possible sensor locations yields the most information. However, if the total number of sensors that could be used is limited to two, than it is clearly most beneficial for locations 1 and 3 to be used, as opposed to locations 1 and 2. This is because sensors at locations 1 and 3 can still yield significant information about component 2 in addition to yielding more detailed information about component 1.

Example 23. Expected information utility for various sensor placement arrangements for a continuous time-based system.

Consider the same binary-state on-demand system illustrated in Figure 49 in Example 22, however the system is now continuous, time based. Only one sensor can be placed on the system: sensor #1 location (for a cost of \$1 000) or sensor #2 location (for a cost of \$ 100). Component A has a constant failure rate, λ_1 . Components B and C have a constant failure rate, λ_2 . Component D's time to failure is described by a normal distribution with mean μ_3 and standard deviation σ_3 . The set of unknowns of interest is listed in equation (225).

$$\boldsymbol{\theta} = \{\lambda_1, \lambda_2, \mu_3, \sigma_3\} \quad \text{---(225)}$$

Prior information consists of limits on the parameters in (226), (227) and (228); and the Bayesian inference of a single test with evidence in equation (229).

$$\lambda_1 \text{ and } \lambda_2 \sim [0,10] \dots \mu_3 \sim [10,15] \dots \sigma_3 \sim [0,5] \quad \text{---(226)(227)(228)}$$

$$\hat{E} = \{\hat{t}_1^S = 0.8, \hat{t}_2^S = 0.7, \hat{t}_3^S = 0.5\} \dots \text{noting that the timing uncertainty is } \Delta t = 0.1 \quad \text{---(229)}$$

The resulting state of knowledge yields has marginal distributions for the unknowns of interest illustrated in Figure 50.

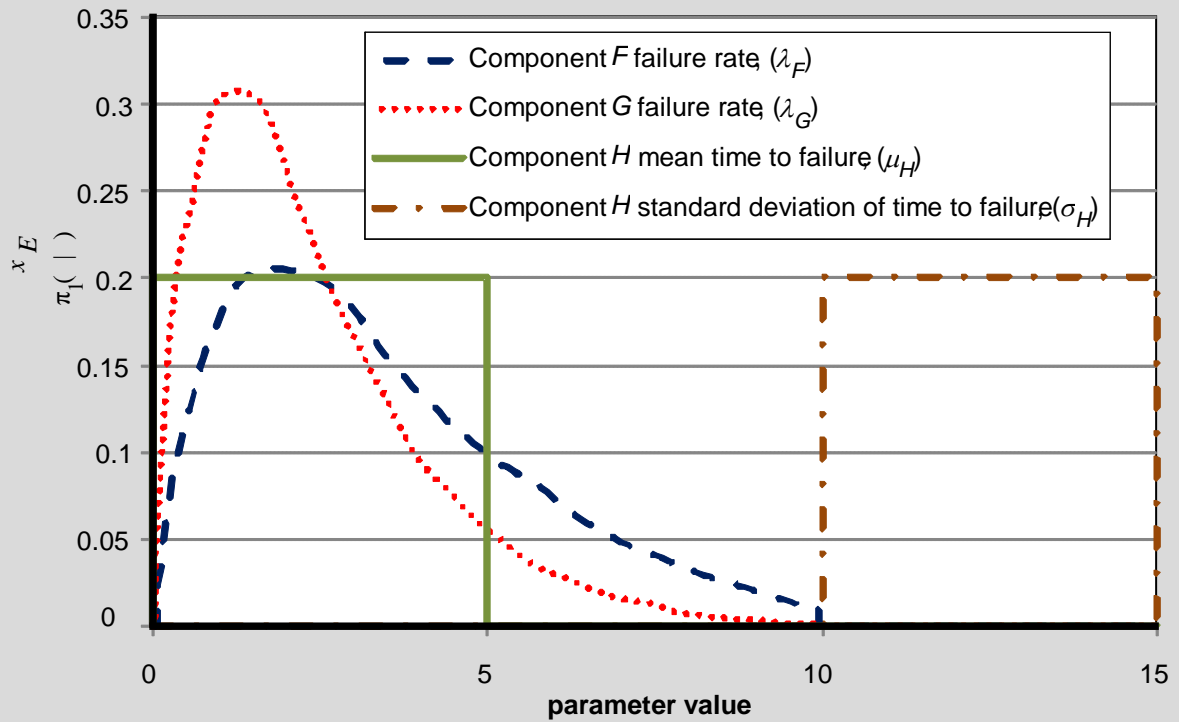


Figure 50: Marginal distributions of the unknowns of interest generated by our current state of knowledge (prior distributions from equations (226), (227), (228) and evidence set (229)).

The same state of knowledge yields the following time to failure distributions for each component, illustrated in Figure 51.

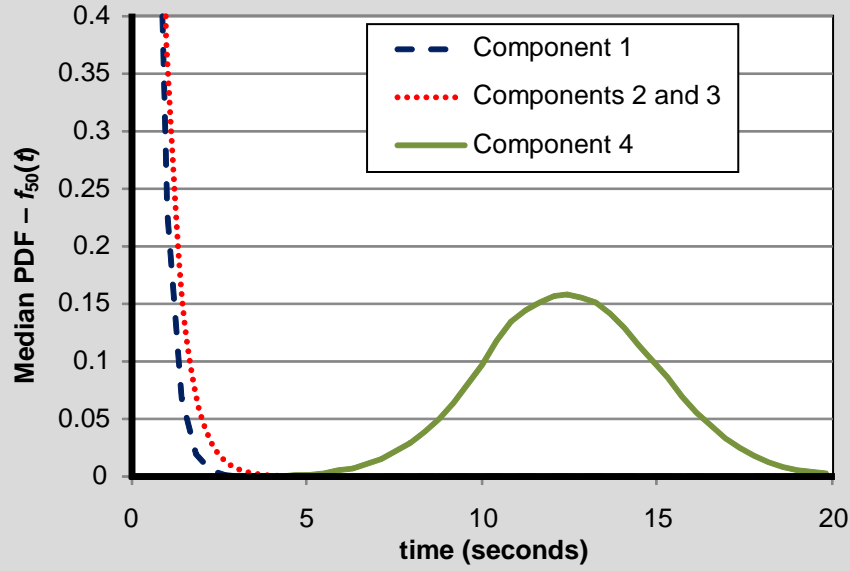


Figure 51: Time to failure distributions for components 1, 2, 3 and 4 based on the state of knowledge represented in Figure 50.

The information utility used to assess the value of various sensor placements is selected to be the inverse of the sum of the posterior variance of all parameters (row three of Table 21):

$$U^I(\varepsilon, E | \pi_0) = U^I(\pi_1 | \pi_0) = \frac{1}{\sum_{j=1}^n \sigma_{\theta_j}^2} \dots \text{ where } \boldsymbol{\theta} \sim \pi_1(\boldsymbol{\theta} | E, \varepsilon) \quad \text{---(230)}$$

The possible sensor arrangements are listed in equations (231) and (232) .

$$\varepsilon_1 = \{1\} \dots \text{(i.e. systemic level failure detection only)} \quad \text{---(231)}$$

$$\varepsilon_2 = \{2\} \dots \text{(i.e. sensor \#2 failure detection)} \quad \text{---(232)}$$

From chapter 7, the likelihood function for this on-demand system is expressed below.

$$\begin{aligned} L(\hat{\mathbf{t}}^S | \boldsymbol{\theta}, \Delta t) &= L(\{\hat{t}_1^S, \dots, \hat{t}_m^S\} | \tilde{t}_1, \dots, \tilde{t}_n, \Delta t) \\ &= \prod_{i=1}^m \Pr(\hat{t}_i^S \leq T_i^S < (\hat{t}_i^S + \Delta t) | \tilde{\boldsymbol{\theta}}_i^{\subset S}, \hat{\mathbf{t}}_i^{\subset S}) \\ &\equiv \prod_{i=1}^m \left[F_i^S(\hat{t}_i^S + \Delta t | \tilde{\boldsymbol{\theta}}_i^{\subset S}, \hat{\mathbf{t}}_i^{\subset S}) - F_i^S(\hat{t}_i^S | \tilde{\boldsymbol{\theta}}_i^{\subset S}, \hat{\mathbf{t}}_i^{\subset S}) \right] \\ &= \prod_{i=1}^m \left[p_i^S \Big|_{t=\hat{t}_i^S + \Delta t} - p_i^S \Big|_{t=\hat{t}_i^S} \right] \end{aligned} \quad \text{---(178)}$$

where Δt is the time increment used in measuring time to failure detection, $\hat{\mathbf{t}}^S$ (as opposed to $\tilde{\mathbf{t}}^S$) represents the vector of failure detection times with an inherent uncertainty, Δt , and $\hat{\mathbf{t}}_i^{\subset S}$ is the set of all uncertain failure detection times of sensors that are *inferentially subordinate* to the i^{th} sensor (i.e. the failure detection times of all sensors $\in \mathbf{S}_i^{\subset S}$).

The structure functions for each possible sensor location remain unchanged from Example 22 where failure probabilities can be used in place of state variables (not all structure functions can use failure probabilities, but those in Example 22 are constructed to allow this to happen).

Using Monte-Carlo simulation, the set $\{\lambda_1, \lambda_2, \mu_3, \sigma_3\}$ is drawn a large number of times based on the state of knowledge that is represented by marginal distributions illustrated in Figure 50.

Each draw defines the PDF and CDF of each component. This allows *component failure times* and hence *sensor failure detection times*, \hat{t}_i^S , to be randomly drawn again. This process yields the simulated sensor detection times for each sensor placement arrangement in Table 26, using an assumed time measurement accuracy of $\Delta \hat{t} = 0.5$.

| $\varepsilon_1 = \{1\} \dots$ Sensor # 1 | | $\varepsilon_2 = \{2\} \dots$ Sensor # 2 | |
|--|---|--|---|
| Failure Detection Time | $\Pr(\hat{t}_1^S \varepsilon, \pi_0) \approx$ | Failure Detection Time | $\Pr(\hat{t}_2^S \varepsilon, \pi_0) \approx$ |
| $0 \leq \hat{t}_1^S < 0.5$ | 0.4373 | $0 \leq \hat{t}_2^S < 0.5$ | 0.4369 |
| $0.5 \leq \hat{t}_1^S < 1.0$ | 0.2517 | $0.5 \leq \hat{t}_2^S < 1.0$ | 0.251 |
| $1.0 \leq \hat{t}_1^S < 1.5$ | 0.1172 | $1.0 \leq \hat{t}_2^S < 1.5$ | 0.1172 |
| $1.5 \leq \hat{t}_1^S < 2.0$ | 0.0618 | $1.5 \leq \hat{t}_2^S < 2.0$ | 0.0618 |
| $2.0 \leq \hat{t}_1^S < 2.5$ | 0.0363 | $2.0 \leq \hat{t}_2^S < 2.5$ | 0.0364 |
| ... | ... | ... | ... |

Table 26: Simulated failure detection times based on current state of knowledge (five most probable evidence sets displayed ... based on 100 000 simulations)

Each simulated evidence set along with equation (178) can be substituted into equation (188) to yield a posterior distribution. The utility function for each posterior distribution is given in equation (230). Equation (193) can then be used to generate the expected utility of each sensor placement arrangement, and the data is summarized in Table 27.

| ε_i | Expected Information Utility, $\bar{U}^I(\varepsilon_i \pi_0)$ |
|-----------------|--|
| {1} | 1.141 |
| {2} | 1.128 |

Table 27: Expected information utility for each sensor arrangement.

It can be seen that the expected utility for each sensor is very similar. This is because based on our current state of knowledge, component 4 will almost certainly fail after all of the other components (as illustrated in Figure 51). Therefore, sensors at locations 1 and 2 will almost certainly detect failure at the same time due to the system logic. The sensor at location 1 has higher expected information utility as it is predicated on the nature of component 4, and in effect is improving the understanding of the component 4 failure characteristics as it will suggest that component 4 hasn't failed when sensor #2 would otherwise detect failure. The sensor at location 2 yields information directly onto the remaining 3 components only.

Notwithstanding the slight difference in information utility, the cost of installing a sensor at the second possible location is significantly less than that for the first possible location. Therefore, it is probably most valuable (in both an information utility and cost/benefit perspective) to install the sensor at the second possible location.

8.7. SUMMARY

Optimizing experimental design is a well documented and researched topic, and is used in a wide array of applications such as dynamic systems to neuroscience. [34, 35] The ability to maximize the expected gain of information through constraining the data gathering process in specific ways is generally desirable. With concepts developed in this dissertation that allow Bayesian analysis of *overlapping data* drawn from systems with multiple sensors, the concept of experimental design (often associated with research and development) can extend to the act of sensor placement (often associated with operational use).

Presented in this chapter is a method of measuring the information utility of various sensor placement arrangements in a Bayesian construct of both on-demand and time based continuous systems. Prior information is used to simulate evidence sets, which are then used to simulate posterior distributions. Information utility is derived from these posterior distributions, and an expected information utility can then be attributed to sensor placement. An example was generated that highlighted this process.

Chapter 9: Conclusion

The fundamental problem that this dissertation addresses is the reliability analysis of multiple *overlapping data sets* that are simultaneously drawn from the same system or process.

9.1. OVERLAPPING-DATA

It is often the case that higher level data sets that are used to assess lower level parameters of systems will be *overlapping* or dependent in nature. For instance, if the data is derived at various component, sub-system and assembly levels from the same system at the same time, they are *overlapping*. Furthermore, there may be relationships between higher level data and multiple manifestations of the same lower level parameter. Such instances demand special considerations and concepts in order to fully infer all aspects of available information. Chapter 1 explains why data sets from the same system drawn simultaneously are fundamentally different to their non-simultaneous counterparts through introducing and examining the concept of *overlapping data*. Sets of *overlapping data* need to meet the following criteria:

- a. Simultaneity - the data sets occur at the same time; and
- b. Correspondence – the data sets are resultant from the same system or process.

Chapter 2 outlined how in the context of Bayesian analysis of data sets drawn from multiple sensors from the same system, the majority of previous techniques centred around on-demand systems and were unable to take into consideration *overlapping data sets*. Whilst fully Bayesian

methodologies have been developed to incorporate data at various levels within on-demand systems, all but one has been constrained to treat all data as *non-overlapping*. The single latter technique is limited to binary-state on-demand systems and is prescriptive in terms of the type of data it can draw. Treating *overlapping data* as *non-overlapping* ignores the dependencies between the data sets and effectively removes inherent information.

9.2. OVERLAPPING DATA ANALYSIS

Chapters 3 to 7 developed methodologies to analyse *overlapping data* on various system scenarios. Several examples were developed to highlight the effect of the additional information *overlapping data* sets contain and how it can be used to correctly improve the state of knowledge (which is the set of component reliability characteristics). The flexibility of the likelihood functions were also generalised to incorporate multiple instances of the same component occurring in the system. Through state of knowledge dependence, the resultant Bayesian *overlapping data* method completely incorporates all information and evidence that can possible be generated or observed by complex systems.

9.2.1. Binary-state on-demand systems

Chapter 3 dealt with *overlapping data* analysis of binary-state on-demand systems. On-demand systems, where components are considered to be either ‘functional’ or ‘failed’, had a likelihood function developed to allow incorporation of *overlapping data* into Bayesian analysis.

The likelihood function in equation (53) allows complete generalisation of the evidence: all that is needed is the total number of demands the system is subjected to, and the total number of failures observed at each sensor. This generalisation separates it from all other techniques and makes it the most flexible.

9.2.2. Multi-state on-demand systems

Chapter 4 covered a methodology to deal with *overlapping data* from multi-state on-demand systems. Multi-state on-demand systems where components are considered to be ‘functional’, ‘failed’, or one of any number of degraded states in between utilise the following likelihood function for Bayesian analysis of overlapping higher-level data. The same property that is observed in the likelihood function for binary-state on-demand systems (that of evidence generalisation) is also observed in equation (64). The evidence can be of the form of total demands the system is subjected to, and the total numbers of particular states observed at each sensor.

9.2.3. Overlapping data analysis of on-demand systems: Algorithm

Chapter 5 outlined the algorithm that rapidly solves the likelihood functions in equations (53) and (64). Both functions revolve around the generation of combinations of component state vectors that match the evidence and subsequently calculate the sum of the probabilities of each. The number of possible combinations that must be considered increases exponentially as the number of components and possible states increases, significantly affecting computational time.

The algorithm instead considers *sensor information vectors* instead of component state vectors, By doing this, it exploits the fact that there will always be fewer sensors than components, and hence fewer combinations to consider. Sets of constraints and limits are generated, thereby creating a rule based of systemic rejection of each *sensor information vector*. An example was developed where the algorithm simplified the solution of the likelihood function by being able to consider 9 different combinations of 4 *sensor information vectors* as opposed to the completely trivial method of individual consideration of a possible 3 724 680 960 combinations of 10.

9.2.4. Continuous Life Metric Systems

Continuous time-based systems were considered in chapter 6 where components whose time to failure is a random variable utilise the following likelihood function for Bayesian analysis of overlapping higher-level data. Whilst the term ‘time-based’ was used, the methodology is equally applicable to a system with any continuous life metric (such as distance). The methodology is based on considering each sensor in isolation, where subordinate sensor data is used to contextualise the failure data of each sensor in question.

9.3. BAYESIAN ANALYSIS OF UNCERTAIN DATA

Chapter 7 dealt with generalising the likelihood functions developed above to incorporate uncertain data, thereby realising several computational efficiencies. Several existing methodologies for Bayesian analysis of uncertain data were examined. Each methodology relied on a probabilistic relationship between observed and actual evidence. This relationship (or error) involved this conditional probability distribution of observed on actual evidence, or vice versa.

9.3.1. On-demand systems

This examination found that the likelihood functions in equations (53) and (64) inherently involve classical error in their derivation, which is the conditional probability of observing a specific evidence set based on the actual evidence set.

9.3.2. Continuous time-based systems

The examination of Bayesian analysis of uncertain data for continuous time-based systems allowed exploitation of the inherent inaccuracies that always exist in measurement devices (such as the accuracy of a clock or timer). The likelihood function in equation (178) is based on uncertain data for continuous time-based systems that not only reflects the reality of inherent timing inaccuracies for all timing devices, but is much more computationally efficient.

9.4. SENSOR PLACEMENT: MAXIMISING INFORMATION FROM BAYESIAN ANALYSIS OF COMPLEX SYSTEMS

At the heart of Bayesian analysis is the concept of updating or improving state of knowledge or information. This is formally implemented by observing a process or system, gathering evidence and incorporating it through Bayes' Theorem to modify a joint distribution of the 'unknowns of interest. Typically, improving information through Bayesian analysis is in itself the only desired outcome. However, in some instances it may be beneficial to quantify the amount of information 'improvement'. Reliability data can be gathered by placing sensors at various levels or places throughout the system (e.g. at all sub-systems), but there are practical and resource limitations to

the number and locations of these sensors. It may be beneficial to assess the probable improvement in information based on various permutations of sensor placement.

9.4.1. Bayesian Experimental Design

Bayesian Experimental Design provides a framework through which permutations of sensor locations can be assessed against the expected value of a *utility function*. The *utility function* defines the ‘worth’ or ‘value’ of a particular permutation in a theoretical probabilistic Bayesian construct. Bayesian inference allows the *utility function* to incorporate both the meaning or information contained by observing a given set of evidence and any prior information that exists about the parameters or the unknowns of interest. This allows a robust approach to be taken to sensor placement in complex systems that incorporate implied information within *overlapping data* sets, permitting the engineer to make an informed decision about sensor location.

9.5. FURTHER WORK

The methodologies covered within this dissertation can be developed further in the following ways or fields:

- a. *Overlapping environmental data*. The data considered has been strictly limited to reliability data. It is possible that *overlapping environmental data* (such as temperature or humidity) is gathered, further contextualising the analysis.

- b. Multi-state continuous life metric systems. The methodology developed in chapter 6 is limited to binary-state continuous life metric systems. Further generalisation to multi-state systems has not been developed.

- c. Uncertain data analysis of continuous life metric systems: inconsistent measurement inaccuracies. The methodology considered in chapter 7 where inaccuracies in life metrics are incorporated (such as the tolerance of a digital stopwatch) require these inaccuracies to be uniform. However, further development could allow analysis where the inaccuracies vary across sensors.

- d. Network and chain modelling analysis. Bayesian belief networks (BBN) and Markov Chains were not modelling methodologies considered in this dissertation. These methodologies (which are generally used for multi-state continuous life metric systems) can have specific *overlapping data techniques* developed.

Appendix A

Consider a likelihood function of the form shown in equation (A1).

$$L(E | x) = B + \sum_{i=1}^n A_i \frac{dH(t - x_i)}{dx} \quad \text{---(A1)}$$

where A_i and B are functions of x (for $i = 1 \dots n$), the evidence set $E = \{x_1, x_2, \dots, x_i\}$ and $H(x)$ is the *Heaviside step function* (which is defined as 0 when $x < 0$, and 1 when $x > 0$).

By definition, the derivative of the $H(x)$ is 0 when $x \neq 0$, as $H(x)$ is constant in this region.

Therefore, the terms in the summation in equation (A1) where $x \neq x_i$ can be excluded.

$$\text{i.e.} \quad L(E | x) = B + \sum_{\forall i \in \tilde{i}} A_i \frac{dH(x - x_i)}{dx} \quad \text{---(A2)}$$

where $\tilde{i} = \{\forall i\}$ such that $x = x_i$.

Therefore, if $\tilde{i} = \emptyset$ (i.e. there are no instances where $x = x_i$) or $A_i = 0$ for all $i \in \tilde{i}$, then the likelihood function reduces to:

$$L(E | x) = B \quad \text{---(A3)}$$

Otherwise, more manipulation is required. A centralized definition for a derivative is

$$\frac{df(x)}{dx} = \lim_{\Delta x \rightarrow 0} \frac{f\left(x + \frac{1}{2}\Delta x\right) - f\left(x - \frac{1}{2}\Delta x\right)}{\Delta x} \quad \text{---(A4)}$$

Therefore, equation (A1) can be rewritten as

$$\begin{aligned} L(E|x) &= B + \sum_{\forall i \in \mathbf{i}} \lim_{\Delta x \rightarrow 0} \frac{A_i \left[H\left(x - x_i + \frac{1}{2}\Delta x\right) - H\left(x - x_i - \frac{1}{2}\Delta x\right) \right]}{\Delta x} \\ &= \lim_{\Delta x \rightarrow 0} \left(B + \sum_{\forall i \in \mathbf{i}} \frac{A_i \left[H\left(x - x_i + \frac{1}{2}\Delta x\right) - H\left(x - x_i - \frac{1}{2}\Delta x\right) \right]}{\Delta x} \right) \\ &= \lim_{\Delta x \rightarrow 0} \left(B + \sum_{\forall i \in \mathbf{i}} \frac{A_i \left[H\left(\frac{1}{2}\Delta x\right) - H\left(-\frac{1}{2}\Delta x\right) \right]}{\Delta x} \right) \text{ since } x = x_i \text{ for all } i \in \mathbf{i} \\ &= \lim_{\Delta x \rightarrow 0} \left(B + \sum_{\forall i \in \mathbf{i}} \frac{A_i [1-0]}{\Delta x} \right) \text{ since } H(x) = \begin{cases} 1 & x > 0 \\ 0 & x < 0 \end{cases} \\ &= \lim_{\Delta x \rightarrow 0} \left(B + \sum_{\forall i \in \mathbf{i}} \frac{A_i}{\Delta x} \right) \quad \text{---(A5)} \end{aligned}$$

When evaluated in isolation, equation (A5) is undefined. However, the normalizing factor of Bayes' Theorem can be exploited. Substituting equation (A3) into Bayes' Theorem yields:

$$\pi_1(x|E) = \frac{L(E|x) \pi_0(x)}{\int_{\forall x} L(E|x) \pi_0(x) dx},$$

$$\begin{aligned}
&= \frac{\lim_{\Delta x \rightarrow 0} \left(B \pi \sum_{\forall i \in \tilde{i}} \frac{A_i}{\Delta x} \right) \phi(\cdot)}{\int_{\forall x} \lim_{\Delta x \rightarrow 0} \left(B \pi \sum_{\forall i \in \tilde{i}} \frac{A_i}{\Delta x} \right) \phi(\cdot) dx} \\
&= \frac{\lim_{\Delta x \rightarrow 0} \left((B \bullet \Delta x) \sum_{\forall i \in \tilde{i}} A_i \right) \phi(\cdot)}{\int_{\forall x} \lim_{\Delta x \rightarrow 0} \left((B \bullet \Delta x) \sum_{\forall i \in \tilde{i}} \Delta x A_i \right) \phi(\cdot) dx} \\
&= \frac{\left(\sum_{\forall i \in \tilde{i}} A_i \pi \right) \phi(\cdot)}{\int_{\forall x} \left(\sum_{\forall i \in \tilde{i}} A_i \pi \right) \phi(\cdot) dx} \quad \text{---(A6)}
\end{aligned}$$

It can be seen from inspection of equation (A6) that the likelihood function is now equivalent to the sum of all A_i where $x = x_i$. Therefore, the likelihood function written in equation (A1) can be expressed as equivalence when used in Bayesian Analysis:

$$\text{given } \tilde{i} = \{\forall i\} \text{ such that } x = x_i \dots L(E|x) \equiv \begin{cases} \sum_{\forall i \in \tilde{i}} A_i & \tilde{i} \neq \emptyset \text{ and } \sum_{i \in \tilde{i}} A_i \neq 0 \\ B & \text{otherwise} \end{cases} \quad \text{---(A7)}$$

Bibliography

1. Hickman, J.W., *PRA procedures guide: A guide to the performance of probabilistic risk assessments for nuclear power plants*. NUREG/CR-2300, 1983. **1**.
2. Mastran, D. and N. Singpurwalla, *Incorporating component and system test data into the same assessment: a Bayesian approach*. Operations Research, 1976. **24**(3): p. 491-499.
3. Martz, H., R. Waller, and E. Fickas, *Bayesian reliability analysis of series systems of binomial subsystems and components*. Technometrics, 1988. **30**(2): p. 143-154.
4. Martz, H. and R. Waller, *Bayesian reliability analysis of complex series/parallel systems of binomial subsystems and components*. Technometrics, 1990. **32**: p. 407-416.
5. Johnson, V., et al., eds. *A hierarchical model for estimating the reliability of complex systems*. Bayesian statistics ed. B.M. Bernardo JM, Berger JO, Dawid AP, Heckerman D, Smith AFM, West M. Vol. 7. 2003, Oxford University Press: London. 199-213.
6. Hamada, M., et al., *A fully Bayesian approach for combining multilevel failure information in fault tree quantification and optimal follow-on resource allocation*. Reliability Engineering and System Safety, 2004(86): p. 297–305.

7. Graves, T., et al., *Using simultaneous higher-level and partial lower-level data in reliability assessments*. Reliability Engineering & System Safety, 2008. **93**(8): p. 1273-1279.
8. Barlow, R. and A. Wu, *Coherent Systems with Multi-State Components*. Mathematics of Operations Research, 1978. **3**(4): p. 275-281.
9. Graves, T., et al., *A fully Bayesian approach for combining multi-level information in multi-state fault tree quantification*. Reliability Engineering and System Safety, 2007. **92**: p. 1476-1483.
10. Klamann, R. and A. Koehler. *Large-scale qualitative modeling for system prediction*. in *Proceedings for computational social and organizational systems conference, South Bend, Indiana*. 2005.
11. Martz, H. and R. Almond, *Using higher-level failure data in fault tree quantification*. Reliability Engineering and System Safety, 1997. **56**: p. 29-42.
12. Mosleh, A. and V. Bier, *On decomposition and aggregation error in estimation: some basic principles and examples*. Risk Analysis, 1991. **12**(2): p. 203-214.

13. Azaiez, M. and V. Bier, *Aggregation error in Bayesian estimation of reliability systems*. Management Science, 1996. **42**(4): p. 516-528.
14. Barthorpe, R. and K. Worden, *Sensor Placement Optimization*. Encyclopedia of Structural Health Monitoring, 2009.
15. Watson, J.-P., H. Greenberg, and W. Hart. *A Multiple-Objective Analysis of Sensor Placement Optimization in Water Networks*. in *Proceedings of the World Water and Environment Resources Congress*. 2004. Salt Lake City, UT: American Society of Civil Engineers
16. Dhillon, S. and C. K. K, *Sensor placement for effective coverage and surveillance in distributed sensor networks*. Wireless Communications and Networking, 2003. **3**: p. 1609-1614.
17. Apostolakis, G. and S. Kaplan, *Pitfalls in risk calculations*. Reliability Engineering, 1981. **2**(2): p. 135-145.
18. Kjaerulff, U.B. and A.L. Madsen, *Bayesian Networks and Influence Diagrams: A guide to Construction and Analysis*. Information Science and Statistics, ed. M. Jordan, J. Kleinber, and B. Scholkopf. 2008, New York: Springer.

19. Jeffreys, H., *An Invariant Form for the Prior Probability in Estimation Problems*. Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences, 1946. **186**(1007): p. 453-461.

20. Bolstad, W., *Understanding Computational Bayesian Statistics*. Wileys Series in Computational Statistics. 2010: John Wiley.

21. Carrol, R. and L. Stefanski, *Approximate quasi-likelihood estimation in models with surrogate predictors*. JASA, 1990(September): p. 652-663.

22. Darby, S. and T. Feam, *Berkson error model*, in *Encyclopedia of statistical sciences updates*, S. Kotz, R. CB, and B. DL, Editors. 1999, Wiley: New York.

23. Mallick, B., F. Hoffman, and R. Carroll, *Semiparametric regrssion modeling with mixtures of Berkson and classical error, with application to foallout from the Nevada test site*. Biometrics, 2002. **58**(1): p. 13-20.

24. Jeffrey, R., *The logic of decision*. 1965, Chicago: University of Chicago Press.

25. Cheeseman, P., ed. *Probabilistic versus fuzzy reasoning*. Uncertainty in artificial intelligence, ed. K. L and L. J. 1986, Elsevier: Dordrecht.

26. Tan, Z. and W. Xi, *Bayesian analysis with consideration of data uncertainty in a specific scenario*. Reliability Engineering & System Safety, 2003. **79**(1): p. 17.

27. Fuller, W., *Measurement error models*. 1987, New York: Wiley.

28. Chaloner, K. and I. Verdinelli, *Bayesian Experimental Design: A Review*. Statistical Science, 1995. **10**(3): p. 273-304.

29. Lindley, D., *On a measure of information provided by an experiment*. The Annals of Mathematical Statistics, 1956. **27**(4): p. 986-1005.

30. Lehmann, E. and G. Casella, *Theory of Point Estimation*. 2nd ed. 1998, New York: Springer-Verlag. 590.

31. Shannon, C.E., *A Mathematical Theory of Communication*. The Bell System Technical Journal, 1948. **27**: p. 279-423.

32. Beirlant, J., Dudewicz, E. J., Györfi, L. and van der Meulen, E. C., *Nonparametric entropy estimation: An overview*. International Journal of the Mathematical Statistics Sciences, 1997. **6**: p. 17-39.

33. Steuer, R., *Multiple Criteria Optimization: Theory, Computations and Application*. 1986, New York: John Wiley & Sons, Inc.
34. Eberhart, R. and S. Yuhui. *Tracking and optimizing dynamic systems with particle swarms*. in *Proceedings of the 2001 Congress on Evolutionary Computation*. 2001. Seoul , South Korea
35. Brookes, M., et al., *Optimising experimental design for MEG beamformer imaging* NeuroImage, 2007. **39**(4): p. 1788-1802.